BAB VI
KESIMPULAN DAN SARAN

6.1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa tujuan penelitian untuk membangun Sistem Informasi *Gym* berbasis *website* menggunakan *framework* Laravel telah berhasil dicapai. Sistem yang dikembangkan mampu mengintegrasikan pengelolaan data pelanggan, proses pendaftaran *membership*, pengelolaan jadwal latihan *private trainer*, transaksi produk, pencatatan pembayaran, serta penyusunan laporan transaksi dalam satu sistem. Dengan adanya sistem tersebut, proses operasional *gym* yang sebelumnya dilakukan secara manual menjadi lebih terstruktur, efektif, dan mudah dikelola.

Sistem yang dikembangkan juga berhasil menjawab rumusan masalah penelitian. Proses pendaftaran pelanggan dapat dilakukan secara *online* sehingga memudahkan calon pelanggan dalam melakukan pendaftaran tanpa harus datang langsung ke *gym*. Selain itu, sistem menyediakan pengelolaan jadwal latihan *private trainer* yang lebih terorganisir, pencatatan pembayaran yang terdokumentasi secara terintegrasi, serta laporan transaksi yang dapat digunakan sebagai pendukung pengelolaan operasional dan pengambilan keputusan. Dengan demikian, sistem mampu meningkatkan efisiensi operasional sekaligus mengurangi potensi kesalahan yang sering terjadi pada proses administrasi manual.

Hasil implementasi didukung oleh pengujian menggunakan metode *Black Box Testing* yang menunjukkan bahwa seluruh fungsi utama sistem berjalan sesuai kebutuhan. Selain itu, evaluasi terhadap 43 responden memperoleh nilai rata-rata sebesar 87,90%**,** yang menunjukkan bahwa sistem diterima dengan sangat baik oleh pengguna. Hasil tersebut menunjukkan bahwa sistem mampu memberikan kemudahan dalam pengelolaan data pelanggan, administrasi *gym*, penjadwalan latihan, dan pencatatan pembayaran sehingga tujuan penelitian untuk meningkatkan efektivitas dan efisiensi pengelolaan operasional *gym* telah tercapai.

Penelitian ini memberikan kontribusi berupa tersedianya sistem informasi

gym berbasis *website* yang mendukung digitalisasi proses bisnis *gym* secara terintegrasi. Sistem yang dihasilkan tidak hanya membantu pengelola dalam menjalankan aktivitas operasional, tetapi juga memberikan kemudahan bagi pelanggan dan *private trainer* dalam mengakses layanan sesuai kebutuhan masing-masing. Dengan demikian, penelitian ini menghasilkan solusi yang dapat meningkatkan kualitas layanan operasional *gym* serta menjadi referensi bagi pengembangan sistem informasi *gym* pada penelitian selanjutnya.

6.2. Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat dijadikan sebagai bahan pengembangan lebih lanjut maupun sebagai pertimbangan dalam penerapan sistem. Bagi pengembang, sistem dapat dikembangkan dengan menambahkan fitur notifikasi otomatis melalui email atau WhatsApp untuk mengingatkan jadwal latihan, masa aktif *membership*, dan status pembayaran. Selain itu, sistem dapat diintegrasikan dengan *payment gateway* agar proses pembayaran dapat dilakukan secara otomatis tanpa memerlukan verifikasi manual. Pengembangan aplikasi berbasis Android atau iOS juga dapat dipertimbangkan untuk meningkatkan kemudahan akses pengguna. Di samping itu, pengembangan fitur *booking* jadwal latihan secara *real-time*, presensi menggunakan QR Code, serta penyempurnaan tampilan antarmuka dapat dilakukan untuk meningkatkan kualitas layanan dan pengalaman pengguna.

Bagi pengelola *gym* sebagai pengguna sistem, disarankan untuk memanfaatkan seluruh fitur yang tersedia secara optimal serta melakukan pemeliharaan sistem secara berkala, seperti pembaruan data pelanggan, *membership*, jadwal latihan, dan transaksi agar informasi yang tersimpan tetap akurat dan konsisten. Selain itu, pelatihan bagi administrator maupun pengguna baru juga perlu dilakukan untuk memaksimalkan pemanfaatan sistem sehingga manfaat digitalisasi operasional *gym* dapat dirasakan secara optimal.

DAFTAR PUSTAKA

[1] S. Mufti Prasetiyo, R. Gustiawan, and F. Rizzel Albani, "Analisis Pertumbuhan Pengguna Internet Di Indonesia," vol. 2, no. 1, 2024, [Online]. Available: https://jurnalmahasiswa.com/index.php/biikma

[2] J. U. Keperawatan *et al.*, "HUBUNGAN PERILAKU GAYA HIDUP SEHAT TERHADAP KEBUGARAN JASMANI USIA DEWASA DI FASILITAS OLAHRAGA SUMMIT *GYM* KOTA SEMARANG," vol. 2, no. 2, p. 12, 2022, [Online]. Available: https://ejournal.poltekkes-smg.ac.id/ojs/index.php/JUK

[3] M. Ridwan and Z. Halim, "Perancangan Sistem Informasi Fasilitas *Fitness Gym* Berbasis *Website* Menggunakan *Code*igniter," *Media Online)*, vol. 4, no. 1, pp. 601–609, 2023, doi: 10.30865/klik.v4i1.1039.

[4] I. M. Rahayu and Y. Amrozi, "PERANCANGAN SISTEM INFORMASI FASILITAS *FITNESS* CENTER BERBASIS *WEB* DI UNIVERSITAS ISLAM NEGERI SUNAN AMPEL," *Jurnal SIMETRIS*, vol. 10, no. 2, 2019.

[5] Y. Wahyudin and D. N. Rahayu, "Analisis Metode Pengembangan Sistem Informasi Berbasis *Website*: A Literatur Review," *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi dan Komunikasi*, vol. 15, no. 3, pp. 26–40, Oct. 2020, doi: 10.35969/interkom.v15i3.74.

[6] B. *Web* Studi Kasus Pondok Pesantren Al-Habi Sholeh Kabupaten Kubu Raya and K. Barat Yoki Firmansyah, "Penerapan Metode SDLC *Waterfall* Dalam Pembuatan Sistem Informasi Akademik," 2018.

[7] A. Maulana and M. Sadikin, "Implementasi Sistem Informasi Manajemen Inventaris Berbasis *Web* Di Pusat Teknologi Informasi Dan Komunikasi-BPPT," *Arief Izzuddin/ Setrum*, vol. 7, no. 1, pp. 182–196, 2018.

[8] Z. Hadi Irawan, S. Lestanti, and U. Mawaddah, "RANCANG BANGUN SISTEM INFORMASI MANAJEMEN ADMINISTRASI PADA *GYM* BERBASIS *WEB* MENGGUNAKAN METODE PENGEMBANGAN AGILE," 2024.

[9] R. Ramadhana, F. Pradana, and A. Arwan, "Pengembangan Sistem Informasi

Manajemen Pengelolaan *Member Fitness* dan Kebutuhan Nutrisi *Member* serta Pemantauan Perkembangan Usaha berbasis *Website* (Studi Kasus: Executive *Gym*, Kota Cilegon)," 2021. [Online]. Available: http://j-ptiik.ub.ac.id

[10] P. P. Sistem Informasi Manajemen Keuangan Berbasis *Web* Studi Kasus Karya Swadaya Abadi Happy Anita Margaretha, M. N. Nababan, and H. Anita Margaretha, "SAINTEK (Jurnal Sains dan Teknologi) PERANCANGAN SISTEM INFORMASI MANAJEMEN KEUANGAN BERBASIS *WEB* STUDI KASUS PT. KARYA SWADAYA ABADI".

[11] M. Zawaruddin Abdullah *et al.*, "Rancang Bangun Sistem Informasi Akuntansi Berbasis *Website* menggunakan *Framework* Laravel (Studi kasus pada UKM Batik dan Bordir Desa Pakisaji Kabupaten Malang)," *Jurnal Sains, Teknologi dan Industri*, vol. 18, no. 1, pp. 49–56, 2020.

[12] D. M. Somantri, K. Manaf, B. Subaeki, and Y. Gunawan, "Prosiding Seminar Sosial Politik, Bisnis, Akuntansi dan Teknik (SoBAT) ke-5 Bandung, 28 Oktober 2023 498 RANCANG BANGUN SISTEM INFORMASI PENCATATAN KEUANGAN PADA UMKM STUDI KASUS TOKO LILIS".

[13] O. U. Ariyanto, M. A. I. Pakereng, and T. Wahyono, "Penerapan Teknologi Responsive *Web* Design dan *Framework* Bootstrap Untuk Pembuatan Aplikasi *Web* Reservasi Dokter pada Klinik Pratama Mardi Lestari," 2016.

[14] M. Y. Putra, "Cara sitasi: Putra MY. 2020. Responsive *Web* Design Menggunakan Boostrap Dalam Merancang Layout *Web*," *Information System for Educators and Professionals*, vol. 5, no. 1, pp. 61–70, 2020.

[15] I. A. Alfarisi, A. T. Priandika, and A. S. Puspaningrum, "Penerapan *Framework* Laravel Pada Sistem Pelayanan Kesehatan (Studi Kasus: Klinik Berkah Medical Center)," *Jurnal Ilmiah Computer Science*, vol. 2, no. 1, pp. 1–9, Jul. 2023, doi: 10.58602/jics.v2i1.11.

[16] A. Wijaya, I. Fenriana, L. W. Kusuma, and E. Dwiyanthi Kusuma, "Perancangan Sistem Informasi Pengaduan Masyarakat Berbasis *Web* dengan Menggunakan *Framework* Laravel," *JURNAL ALGOR*, vol. III, no.

2, 2022, [Online]. Available: https://jurnal.buddhidharma.ac.id/index.php/algor/index

[17] I. Gede Handika and A. Purbasari, "Pemanfaatan *Framework* Laravel Dalam Pembangunan Aplikasi E-Travel Berbasis *Website*," Mar. 2018.

[18] F. Sahrul, S. Kom, M. Eng, M. A. Safi'ie, S. Si, and O. Decroly, "'TRANSFORMASI Jurnal Informasi & Pengembangan Iptek'(STMIK BINA PATRIA) IMPLEMENTASI SISTEM INFORMASI AKADEMIK BERBASIS *WEB* MENGGUNAKAN *FRAMEWORK* LARAVEL," 2016.

[19] A. Firman, H. F. Wowor, and X. Najoan, "Sistem Informasi Perpustakaan Online Berbasis *Web*," 2016.

[20] Y. F. Rohman, "Transformasi *Gym*: Antara Olahraga, Simbol, dan Representasi Status Sosial," *Jurnal Socius: Journal of Sociology Research and Education*, vol. 7, no. 1, p. 1, Jun. 2020, doi: 10.24036/scs.v7i1.172.

[21] T. A. KInaswara, N. R. Hidayati, and F. Nugrahanti, "Rancang Bangun Aplikasi Inventaris Berbasis *Website* pada Kelurahan Bantengan," 2019.

[22] A. S. Chechet, M. V. Chernykh, Ia. S. Panasiuk, and I. I. Abdullin, "FRONT-END SECURITY ARCHITECTURE: PROTECTION OF USER DATA AND PRIVACY," *Systems and Technologies*, vol. 68, no. 2, pp. 102–111, Dec. 2024, doi: 10.32782/2521-6643-2024-2-68.12.

[23] A. R. Kusuma, R. Syarief, A. Sukmawati, and A. Ekananta, "Factors influencing the digital transformation of sales organizations in Indonesia," *Heliyon*, vol. 10, no. 5, Mar. 2024, doi: 10.1016/j.heliyon.2024.e27017.

[24] M. Rifqi Fauzan, C. Agustin, P. Studi Sisitem Informasi, and P. Piksi Ganesha, "SISTEM INFORMASI SURAT KELUAR BERBASIS *WEB* MENGUNAKAN XAMPP DAN MySQL DI SD LANGENSARI KOTA CIMAHI," vol. 12, 2024, [Online]. Available: http://journal.piksi.ac.id/index.php/INFOKOM

[25] Y. Sari Siregar, B. Oktaviana Sembiring, E. Rahayu, and R. Franchitika, "Pemanfaatan Aplikasi MySQL untuk Membantu Siswa SMK Swasta Nur Azizi dalam Pengolahan Data," 2024. [Online]. Available: https://jurnal.unity-academy.sch.id/index.php/japamas

LAMPIRAN

Lampiran 1 Dokumen Spesifikasi Kebutuhan Perangkat Lunak

Lampiran 2 Dokumen Deskripsi Perancangan Perangkat Lunak

Lampiran 3 Dokumen Perencanaan, Deskripsi, dan Hasil Uji Perangkat Lunak

Lampiran 4 Dokumen Hasil Kuesioner

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

PEMBANGUNAN SISTEM INFORMASI *GYM* BERBASIS *WEBSITE* MENGGUNAKAN *FRAMEWORK* LARAVEL

Disusun oleh:

Axel Christian Nugroho

210711114

Program Studi Informatika

Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

Daftar Revisi

Tanggal	Deskripsi Revisi	Penulis	Keterangan
<date>	<Version 1>	<Your Name>	<First Revision>

Persetujuan Dokumen

Dokumen Spesifikasi Kebutuhan Perangkat Lunak ini telah diterima dan disetujui oleh:

Nama	Tanggal	Tandatangan

Daftar Isi

Daftar Revisi 112

Persetujuan Dokumen 112

1. Pendahuluan 115

1.1 Tujuan Penulisan Dokumen 115

1.2 Ruang Lingkup 115

1.3 Definisi, Akronim, dan Singkatan 116

1.4 Referensi 116

1.5 Ikhtisar Dokumen 116

2. Deskripsi Umum Kebutuhan 116

2.1 Perspektif Produk 116

2.2 Fungsi Produk 117

2.3 Karakteristik Pengguna 118

2.4 Kekangan 118

2.5 Asumsi dan Kebergantungan 118

3. Kebutuhan Rinci 118

3.1 Kebutuhan Antarmuka Eksternal 118

3.1.1 Antarmuka Pengguna 119

3.1.2 Antarmuka Perangkat Keras 120

3.1.3 Antarmuka Perangkat Lunak 120

3.1.4 Antarmuka Komunikasi 122

3.1.5 Antarmuka Sistem 124

3.2 Kebutuhan Fungsional 125

3.2.1. Menambah Data Produksi 125

3.3 Diagram Use Case.....130
3.4 Kebutuhan Non-Fungsional 138
3.4.1 Performance 138
3.4.2 Reliability 138
3.4.3 Availability 139
3.4.4 Security 139
3.4.5 Maintainability 140
3.5.6 Usability 140
3.5 Kebutuhan Data.....141
A. Lampiran .....**Error! Bookmark not defined.**
A.1 Lampiran 1 .....**Error! Bookmark not defined.**
A.2 Lampiran 2 .....**Error! Bookmark not defined.**

1. Pendahuluan

Dokumen ini berisi spesifikasi kebutuhan perangkat lunak untuk sistem informasi *GYM* berbasis *website* Sistem informasi pengelolaan *membership*, jadwal latihan dengan *private trainer*, pembayaran dan *verifikasi online*, serta laporan transaksi.

1.1 Tujuan Penulisan Dokumen

Dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL) ini disusun untuk *member*ikan gambaran lengkap mengenai kebutuhan fungsional dan bukan fungsional dari Sistem Informasi *GYM* Berbasis *Website* yang dikembangkan menggunakan *framework* Laravel.

Dokumen ini digunakan sebagai acuan dalam tahap perancangan, implementasi, dan pengujian sistem agar sesuai dengan kebutuhan pengguna yang terdiri dari Admin, Pelanggan, dan *Private Trainer*.

1.2 Ruang Lingkup

Sistem Informasi *GYM* berbasis *website* ini mencakup:

a. Pengelolaan data pelanggan
b. Pengelolaan data *private trainer*
c. Pengelolaan paket *membership*
d. Pengelolaan produk (*supplement*, dll)
e. Pengaturan jadwal latihan (agar tidak bentrok)
f. Verifikasi pembayaran *membership* dan produk
g. Riwayat transaksi pelanggan
h. Laporan transaksi dan status kepelangganan
i. Autentikasi dan hak akses berbasis *role* (Admin, pelanggan, *Private Trainer*)

Sistem tidak mencakup:

a. Pembayaran otomatis menggunakan *payment gateway*
b. Sistem latihan berbasis *IoT* atau sensor fisik
c. *Mobile Application* (Android/iOS)

Sistem dikembangkan berbasis *website* responsif dan diakses menggunakan

browser.

1.3 Definisi, Akronim, dan Singkatan

Definisi	
Sistem	Sistem informasi produksi berbasis *website* yang dikembangkan.
GYM	Tempat kebugaran untuk latihan fisik
CRUD	Operasi data dasar(*Create, Read, Update, Delete*).
Pelanggan	Pengguna terdaftar yang memiliki kepelangganan
Private Trainer	Pelatih pribadi untuk pelanggan
Admin	Pengelola utama sistem
Laravel	*Framework* PHP berbasis MVC
ERD	*Entity Relationship Diagram*
MVC	*Model View Controller*
SUS	*System Usability Scale*
CSRF	Perlindungan serangan *web*

1.4 Referensi

1. Laporan, Tugas Akhir Sistem Informasi *GYM* Berbasis *Website*

1.5 Ikhtisar Dokumen

Dokumen SKPL ini disusun untuk mendefinisikan kebutuhan fungsional dan non-fungsional sistem informasi produksi berbasis *website*. Bab-bab selanjutnya memuat deskripsi umum sistem, kebutuhan rinci termasuk antarmuka eksternal, kebutuhan fungsional beserta aturan validasi, use-case, serta batasan, asumsi, dan kebergantungan.

2. Deskripsi Umum Kebutuhan

2.1 Perspektif Produk

Sistem Informasi *GYM* adalah aplikasi berbasis *web* yang dibangun menggunakan Laravel dan *database* MySQL. Sistem ini dapat diakses melalui *browser* dan

digunakan oleh tiga aktor utama:

1. Admin
2. Pelanggan
3. *Private Trainer*

Sistem ini berdiri sendiri dan dapat di-host pada *server* lokal (XAMPP) maupun *server* hosting online.

2.2 Fungsi Produk

Deskripsi	:	Fitur *login*/*logout* dan pengelolaan hak akses berbasis peran untuk Admin, pelanggan, dan *private trainer*.
Rasional	:	Menjamin hanya pengguna berwenang yang dapat mengakses fungsi tertentu sehingga menjaga keamanan data dan mencegah perubahan data yang tidak sah oleh user lain.

Deskripsi	:	Admin mencatat, mengubah, dan menghapus data pelanggan, *private trainer*, paket, produk, dan jadwal.
Rasional	:	Fitur transaksi diperlukan agar proses pembelian *membership*, transaksi produk, dan pengaturan jadwal berjalan terpusat, mudah dipantau, mengurangi kesalahan pencatatan manual, serta meningkatkan efisiensi operasional *gym*.

Deskripsi	:	Pelanggan membeli *membership* atau produk lalu mengunggah bukti pembayaran. Admin memverifikasi pembayaran tersebut.
Rasional	:	Mempermudah proses administrasi dan meminimalkan kesalahan *input*.

Deskripsi	:	Pembuatan Order dan update stok produk secara otomatis.
Rasional	:	Menjamin ketersediaan produk dan akurat sehingga

		mengurangi risiko kekurangan produk.

Deskripsi	:	Halaman *dashboard* berisi laporan penjualan, banyak terjual, dan banyak pelanggan dan modul laporan dengan filter tanggal dan grafik.
Rasional	:	Mempercepat pengambilan keputusan manajerial dan memudahkan analisis historis penjualan dan pelanggan.

2.3 Karakteristik Pengguna

1) Admin: konfigurasi sistem, manajemen user, akses semua fitur administratif.
2) *Private Trainer*: melihat jadwal latihan, mengatur profil.
3) Pelanggan : langganan *membership*, pembelian produk.

2.4 Kekangan

1) Sistem berbasis *web* dan membutuhkan koneksi internet
2) Hanya mendukung Bahasa Indonesia dan mata uang Rupiah
3) Tidak ada integrasi ke sistem eksternal

2.5 Asumsi dan Kebergantungan

1) Pengguna mempunyai akun dan hak akses sesuai peran yang ditetapkan.
2) Sistem akan dibangun menggunakan stack *web* (PHP/Laravel & MySQL)
3) User memiliki perangkat dengan *browser* modern
4) Data awal disediakan oleh admin gym

3. Kebutuhan Rinci

3.1 Kebutuhan Antarmuka Eksternal

Subbagian ini *member*ikan deskripsi rinci seluruh *input* dan *output*. Selain itu, subbagian ini juga menjelaskan antarmuka pengguna, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi dan antarmuka sistem.

3.1.1 Antarmuka Pengguna

Sistem Informasi *GYM* ini memiliki form antarmuka pada tabel dibawah sebagai berikut :

No	Nama Form	:	Deskripsi Fungsi Form
1	Form *Login*	:	Autentikasi pengguna, memasukkan *username*/*password* dan mengarahkan ke *dashboard* sesuai *role*.
2	*Dashboard* (per-*role*)	:	Tampilan menu yang berbeda untuk Admin, *Private Trainer*, dan pelanggan.
3	*Form Input Membership*	:	CRUD *membership*
4	Form Input Produk	:	CRUD produk
5	Form Jadwal Latihan	:	Meng*input* jadwal Latihan dan mencegah bentrok
6	Form Verifikasi Pembayaran	:	Admin verifikasi
7	Form Data *Private Trainer*	:	*Create* dan hapus *private trainer*
8	Form Laporan	:	Filter laporan penjualan

Selain form di atas, Sistem Informasi *GYM* ini juga menyediakan report berikut ini:

No	Nama Report	:	Deskripsi Fungsi Report
1	Laporan Transaksi *Membership*	:	Menampilkan daftar seluruh transaksi pembelian paket *membership* beserta statusnya (Menunggu Verifikasi, Disetujui, Ditolak). Admin dapat memfilter berdasarkan tanggal, nama pelanggan, dan jenis paket.
2	Laporan Transaksi Produk	:	Berisi daftar transaksi pembelian produk *gym* seperti suplemen, termasuk detail pembeli, jumlah produk, harga, status pembayaran, dan tanggal transaksi. Dilengkapi fitur pencarian dan filter.

3	Laporan Pendapatan *GYM*	:	Menggabungkan total pendapatan dari transaksi *membership* dan produk dalam rentang waktu tertentu. Disajikan dalam bentuk tabel dan grafik untuk memudahkan analisis keuangan.
4	Laporan pembelian produk	:	Menampilkan produk yang paling banyak dibeli. Membantu admin mengelola inventori.

3.1.2 Antarmuka Perangkat Keras

Perangkat keras *server* digunakan untuk menjalankan aplikasi Laravel dan *database* MySQL. Berdasarkan lingkungan pengembangan pada *website GYM*, spesifikasi minimum perangkat keras yang diperlukan adalah:

1. **Processor:** Intel Core i5 / Ryzen 5 atau setara
2. **RAM:** Minimal 8 GB (disarankan 16 GB untuk kestabilan *server*)
3. **Storage:** SSD minimal 256 GB
4. **Jaringan:** Koneksi internet stabil (minimal 10 Mbps)
5. **Perangkat *Server*:**
    a. *Server* lokal menggunakan laptop/PC pengembangan
    b. Atau hosting/*web server* berbasis Linux/Windows

Spesifikasi ini sesuai dengan perangkat yang digunakan dalam proses pengembangan, yaitu laptop dengan prosesor Intel i7, RAM 8 GB, dan SSD 1512 GB sebagaimana tertulis pada bagian Setting Penelitian.

3.1.3 Antarmuka Perangkat Lunak

Subbagian ini menjelaskan hubungan antara perangkat lunak dengan perangkat lunak lain yang digunakan dalam pengembangan dan pengoperasian Sistem Informasi *GYM* Berbasis *Website*. Antarmuka perangkat lunak mencakup sistem operasi, *database*, library, *framework*, serta tools pendukung yang berinteraksi dengan aplikasi.

Perangkat lunak yang dibutuhkan untuk mengoperasikan sistem informasi *GYM* (*server*-side) adalah sebagai berikut:

1	Nama	:	PHP
	Sumber	:	https://www.php.net/
	Deskripsi	:	Digunakan sebagai bahasa pemrograman *server*-side utama yang menjalankan aplikasi berbasis Laravel. PHP memproses *request*, menjalankan *controller*, model, dan menghasilkan *output* ke *browser*.
2	Nama	:	Laravel *Framework*
	Sumber	:	https://laravel.com/
	Deskripsi	:	*Framework* PHP yang digunakan untuk membangun arsitektur aplikasi berbasis MVC, menyediakan fitur *routing*, *Eloquent* ORM, *autentikasi*, *middleware*, migrasi *database*, dan keamanan (CSRF, hashing). *Framework* inti dari sistem *GYM*.
3	Nama	:	MySQL *Database Server*
	Sumber	:	https://www.mysql.com/
	Deskripsi	:	DBMS yang digunakan untuk menyimpan seluruh data seperti pelanggan, *private trainer*, paket, jadwal latihan, transaksi, dan bukti pembayaran. Mendukung relasi dan integritas data.
4	Nama	:	Apache *Web Server* (XAMPP)
	Sumber	:	https://www.apachefriends.org/
	Deskripsi	:	*Web server* yang menangani *request* HTTP dari *client* dan meneruskannya ke Laravel. Digunakan di tahap pengembangan lokal (*local development*). Bagian dari paket XAMPP yang dijelaskan pada laporan TA. (Sumber: Setting Penelitian)
5	Nama	:	Git

Sumber	:	https://git-scm.com/
Deskripsi	:	*Version control system* yang digunakan untuk mengelola versi kode, melakukan commit, *tracking* perubahan, dan kolaborasi proyek. *Repository* dapat disimpan di GitHub.

Perangkat lunak yang digunakan oleh pengguna (Admin, Pelanggan, dan *Private Trainer*) untuk mengakses sistem adalah sebagai berikut:

1	Nama	:	*Web Browser* Modern
	Sumber	:	Google Chrome : https://chrome.google.com Mozilla Firefox : https://mozilla.org Microsoft Edge : https://microsoft.com/edge
	Deskripsi	:	Digunakan untuk mengakses aplikasi *web GYM* melalui protokol HTTP/HTTPS. *Browser* menjalankan antarmuka pengguna (UI) berbasis HTML, CSS, dan JavaScript yang dirender oleh Laravel *Blade Template*.
2	Nama	:	Sistem Operasi Klien (Windows / Android / iOS)
	Sumber	:	Windows : https://www.microsoft.com Android : https://www.android.com iOS : https://www.apple.com
	Deskripsi	:	Sistem operasi yang digunakan oleh klien untuk menjalankan *browser*. Karena sistem bersifat *web-based* dan responsif, perangkat apa pun (Windows, MacOS, Android, iOS) dapat mengakses aplikasi selama mendukung *browser* modern.

3.1.4 Antarmuka Komunikasi

A. Protokol Komunikasi

Sistem menggunakan protokol sebagai berikut:

1. HTTP (*HyperText Transfer Protocol*)

Digunakan untuk komunikasi dasar antara klien (*browser*) dan *server* aplikasi Laravel.

2. HTTPS (*HyperText Transfer Protocol Secure*)

Digunakan untuk komunikasi terenkripsi, memastikan keamanan data saat dikirim antara klien dan *server*. HTTPS direkomendasikan terutama untuk proses *login*, upload bukti pembayaran, dan pengolahan data sensitif.

B. Model Komunikasi *Client–Server*

Aplikasi memanfaatkan model komunikasi berikut:

1. *Client* mengirim request melalui *browser* (*GET, POST, PUT, DELETE*).
2. *Server* Laravel memproses permintaan melalui *routing* dan *controller*.
3. *Server* mengakses *database* MySQL bila diperlukan.
4. *Server* mengembalikan *response* dalam bentuk HTML, JSON, atau tampilan Blade.

C. Pertukaran Data

Jenis data yang ditransfer antara klien dan *server*:

1. Data teks (*input form* pelanggan, pembayaran, jadwal)
2. Data file (upload bukti pembayaran berbentuk gambar/PDF)
3. Data JSON (digunakan pada AJAX untuk cek bentrok jadwal)

Contoh komunikasi internal:

Sistem mengirim request AJAX untuk memeriksa bentrok jadwal latihan sebelum menyimpan jadwal.

D. Media Komunikasi

Sistem berbasis *web* menggunakan jaringan berikut:

1. Internet / Intranet

Untuk akses aplikasi oleh pelanggan dan *Private Trainer* dan akses admin terhadap *dashboard* dan laporan

2. Jaringan Lokal (*Localhost*)

Digunakan pada tahap pengembangan menggunakan XAMPP (Sumber: *Setting* Penelitian)

E. Keamanan Komunikasi

Untuk menjaga keamanan pertukaran data, sistem menerapkan:

1. CSRF Token untuk mencegah serangan *Cross-Site Request Forgery*
2. Hashing *Password* menggunakan bcrypt/argon2
3. Validasi *input* pada sisi *server*
4. HTTPS (di *server* produksi)

F. Kompatibilitas Komunikasi

Sistem kompatibel dengan:

1. *Browser* yang mendukung protokol HTTP/HTTPS
2. Perangkat *mobile* (Android/iOS)
3. Perangkat *desktop* (Windows/macOS/Linux)

Tidak diperlukan instalasi perangkat lunak tambahan pada sisi klien.

3.1.5 Antarmuka Sistem

Subbagian ini menjelaskan hubungan Sistem Informasi *GYM* dengan sistem lain yang mungkin terintegrasi secara langsung maupun tidak langsung. Berdasarkan analisis kebutuhan dan batasan ruang lingkup penelitian, sistem ini tidak melakukan integrasi dengan sistem *eksternal* apa pun.

Sistem beroperasi secara mandiri dan tidak memerlukan pertukaran data dengan aplikasi atau layanan pihak ketiga.

1. Integrasi dengan Sistem Lain

   Sistem Informasi *GYM* tidak memiliki integrasi dengan:

   a. Sistem informasi pembayaran eksternal (*payment gateway*)
   b. Sistem kehadiran/absensi berbasis perangkat keras
   c. Sistem manajemen *gym* lain
   d. Sistem *IoT* atau sensor latihan
   e. *Database eksternal* milik pihak ketiga

Hal ini sesuai dengan batasan penelitian yang menyatakan bahwa sistem hanya menyediakan pembayaran melalui unggah bukti pembayaran, bukan integrasi API otomatis.

2. Antarmuka Sistem

   Karena tidak ada sistem eksternal yang terhubung, maka antarmuka sistem tidak diperlukan.

   Seluruh proses terjadi secara internal dalam aplikasi Laravel, yang meliputi:

   a. Pengelolaan data (pelanggan, *private trainer*, produk, *membership*)
   b. Pengaturan jadwal
   c. Transaksi pembayaran manual
   d. Generasi laporan

   Tidak ada API eksternal, *web service*, atau modul integrasi yang digunakan pada sistem ini.

3.2 Kebutuhan Fungsional

Subbagian ini menjelaskan kebutuhan fungsional perangkat lunak secara rinci. Setiap kebutuhan diberi nomor unik untuk memudahkan proses penelusuran dalam perancangan, implementasi, dan pengujian perangkat lunak.

Kebutuhan fungsional berikut disusun berdasarkan aktor utama dalam sistem: Admin, Pelanggan, dan *Private Trainer*.

3.2.1. Menambah Data Produksi

ID Requirement	:	FR-*GYM*-001
Deskripsi	:	Form untuk menambah data user baru ke dalam sistem. Data yang dicatat mencakup: nama lengkap, email, nomor telepon, alamat, serta informasi tambahan yang diperlukan untuk proses *membership*. Setelah submit, data disimpan dalam tabel *users* dan dapat digunakan untuk transaksi *membership* serta penjadwalan latihan.

Vality check	:	1. *Username* tidak boleh kosong 2. Nomor telepon harus berupa angka 3. Harus memilih salah satu *Role* 4. *Password* harus minimal 8 karakter 5. Alamat harus terisi 6. Status *membership* diset menjadi "Belum Aktif" sebelum verifikasi pembayaran
Rasional	:	Menambah data pelanggan merupakan fungsi fundamental yang memastikan identitas setiap pengguna tercatat secara akurat. Validasi yang tepat mencegah duplikasi data, memastikan integritas *database*, dan mendukung proses lanjutan seperti transaksi *membership*, upload bukti pembayaran, serta jadwal latihan.
Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel

ID Requirement	:	FR-*GYM*-002
Deskripsi	:	Admin dapat membuat paket *membership* seperti 1 bulan, 3 bulan, 6 bulan, dan 12 bulan yang berisi durasi, harga *male*, harga *female* dan harga *couple*.
Vality check	:	1. Durasi harus memilih antara 1 bulan, 3 bulan, 6 bulan, dan 1 tahun 2. Harga > 0.
Rasional	:	Keberadaan paket diperlukan agar pelanggan dapat memilih layanan sesuai kebutuhan.
Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel

ID	:	FR-*GYM*-003

Requirement		
Deskripsi	:	Form untuk menambah produk suplemen
Vality check	:	1. Nama produk tidak boleh kosong 2. Harga dan stok > 0. 3. Gambar produk harus ada
Rasional	:	Digunakan untuk transaksi penjualan produk *gym*.
Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel

ID Requirement	:	FR-*GYM*-004
Deskripsi	:	Pelanggan memilih paket dan melakukan pembelian dengan mengunggah bukti pembayaran.
Vality check	:	1. Paket harus tersedia. 2. Bukti pembayaran harus berupa file JPG/PNG/PDF 3. Nominal pembayaran harus sesuai harga paket
Rasional	:	Memungkinkan proses pembelian *membership* secara online.
Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel

ID Requirement	:	FR-*GYM*-005
Deskripsi	:	Pelanggan memilih produk dan mengunggah bukti pembayaran.
Vality check	:	1. Produk harus ada dan stok > 0 2. Bukti pembayaran format valid 3. Jumlah produk minimal 1. 4. Nominal pembayaran harus sesuai harga yang ingin dibeli
Rasional	:	Digunakan dalam sistem penjualan produk *gym*.
Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel.

ID Requirement	:	FR-*GYM*-006
Deskripsi	:	Pelanggan memilih produk dan mengunggah bukti pembayaran.
Vality check	:	1. Produk harus ada dan stok > 0 2. Bukti pembayaran format valid 3. Jumlah produk minimal 1. 4. Nominal pembayaran harus sesuai harga yang ingin dibeli
Rasional	:	Digunakan dalam sistem penjualan produk *gym*.
Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel.

ID Requirement	:	FR-*GYM*-007
Deskripsi	:	Admin memverifikasi bukti pembayaran, menerima atau menolak transaksi.
Vality check	:	1. Bukti pembayaran harus ada 2. Nominal harus sesuai 3. Jika diterima → *membership* aktif otomatis
Rasional	:	Menghindari kesalahan pencatatan transaksi dan kecurangan pembayaran.
Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel.

ID Requirement	:	FR-*GYM*-08
Deskripsi	:	pelanggan dan *private trainer* dapat melihat jadwal latihan masing-masing
Vality check	:	1. User harus *login* 2. Pelanggan wajib memilihi paket *bundle* 3. Menampilkan hanya jadwal terkait user
Rasional	:	Memudahkan koordinasi antara pelanggan dan *private trainer*.

Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel.

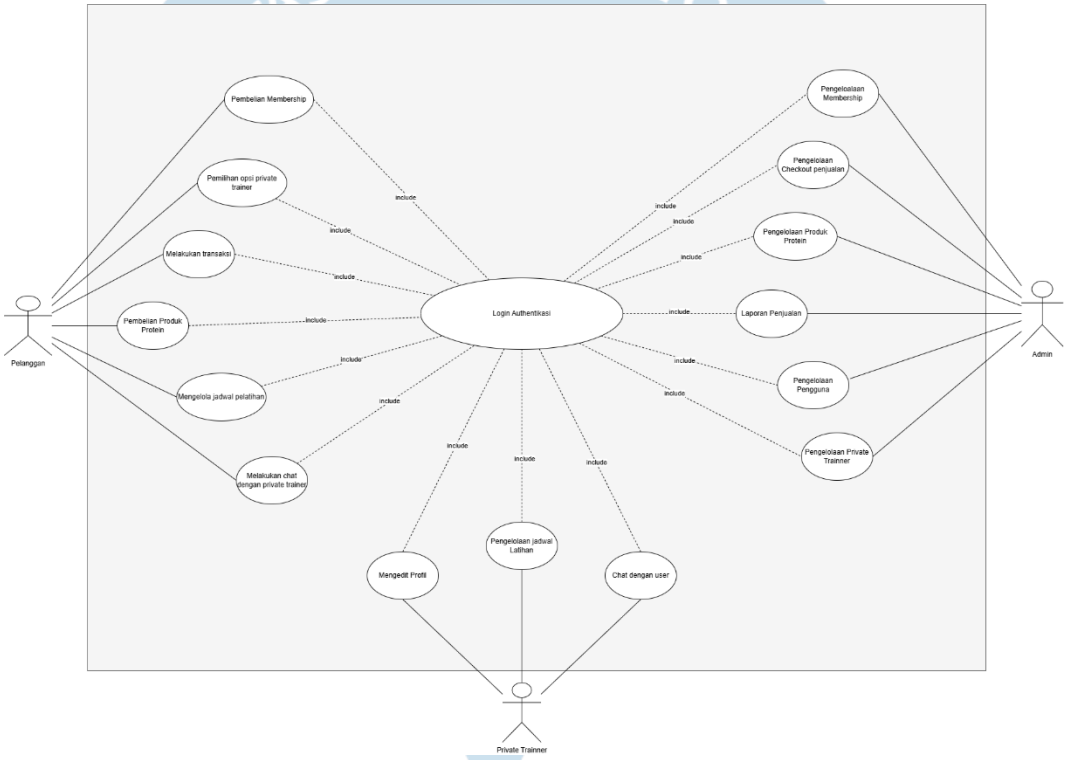
ID Requirement	:	FR-*GYM*-09
Deskripsi	:	Pelanggan melihat riwayat transaksi *membership* atau produk.
Vality check	:	1. User *login* 2. Data ditampilkan berdasarkan ID pelanggan
Rasional	:	Pelanggan mengetahui status pembayaran dan riwayat transaksi
Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel.

ID Requirement	:	FR-*GYM*-12
Deskripsi	:	Admin dapat menghasilkan laporan transaksi *membership* dan produk.
Vality check	:	1. User *login* 2. Filter tanggal harus valid
Rasional	:	Admin mengetahui laporan hasil penjualan dan mengetahui penjualan dari produk dan data *membership* yang banyak dipilih pelanggan
Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel.

ID Requirement	:	FR-*GYM*-13
Deskripsi	:	Semua pengguna dapat memperbarui informasi profil mereka.
Vality check	:	1. *Username* unik 2. Nomor telepon angka 3. *Password* minimal 6 karakter

		4. Alamat harus terisi
Rasional	:	*User* dapat memperbarui data apabila ada perubahan maupun ingin diubah
Referensi	:	Laporan Tugas Akhir Sistem Informasi *GYM* Berbasis *Website* Menggunakan Laravel.

3.3 Diagram Use Case



Identifikasi	:	UC-001
Nama Use Case	:	*Login* / *Logout*
Deskripsi singkat	:	Pengguna melakukan autentikasi untuk mengakses fungsi sesuai peran; dapat *logout* untuk mengakhiri sesi.
Aktor	:	Admin, pelanggan, *Private Trainer*
Pre kondisi	:	Pengguna sudah terdaftar (akun ada di DB) atau berada di

		halaman *login*.
Pos Kondisi	:	Jika sukses → pengguna terautentikasi dan diarahkan ke *dashboard* sesuai *role*; jika *logout* → sesi berakhir.
Basic Flow	:	1. Pengguna membuka halaman *login*. 2. Pengguna memasukkan *username* dan *password*. 3. Sistem memvalidasi kredensial (hash *password*, CSRF token). 4. Jika valid, sistem membuat session dan mengarahkan ke *dashboard role appropriate*.
Alternative Flow	:	Pengguna memilih "Lupa *Password*" → alur pemulihan/forget-*password* (email reset).
Eror Flow	:	menampilkan pesan "*Username* atau *password* salah".

Identifikasi	:	UC-002
Nama Use Case	:	Registrasi pelanggan
Deskripsi singkat	:	Calon pelanggan mengisi form pendaftaran untuk membuat akun.
Aktor	:	Calon User
Pre kondisi	:	Calon user berada di halaman registrasi
Pos Kondisi	:	user terautentikasi dan diarahkan ke halaman *login*.
Basic Flow	:	1. Calon pelanggan mengisi *username*, *password*, no. telepon, alamat, tanggal, memilih *role*. 2. Sistem memvalidasi *input* (*username* unik, nomor telepon angka, alamat harus terisi, *password* ≥6).
Alternative Flow	:	User harus mengisi semua form yang ditentukan
Eror Flow	:	1. *Username* harus unik 2. Nomor telepon harus berupa angka 3. Alamat harus terisi 4. *Password* minimal 6

Identifikasi		UC-501-01
Nama Use Case	:	Kelola Data pelanggan
Deskripsi singkat	:	Admin menghapus, dan melihat data pelanggan.
Aktor	:	Admin
Pre kondisi	:	Admin terautentikasi.
Pos Kondisi	:	Perubahan tersimpan pada table user dengan *role* pelanggan
Basic Flow	:	Menampilkan data user dengan *role* pelanggan
Alternative Flow	:	Admin memilih pelanggan → edit → simpan perubahan.
Eror Flow	:	Mengubah data tidak sesuai ketentuan

Identifikasi		UC-101
Nama Use Case	:	Kelola Data *Membership*
Deskripsi singkat	:	Admin menambah, mengubah, menghapus, dan melihat data *membership*.
Aktor	:	Admin
Pre kondisi	:	Admin terautentikasi.
Pos Kondisi	:	Perubahan tersimpan pada table *membership*
Basic Flow	:	Menampilkan data *membership*
Alternative Flow	:	Admin memilih *membership* → edit → simpan perubahan.
Eror Flow	:	Mengubah data tidak sesuai ketentuan

Identifikasi		UC-101
Nama Use Case	:	Kelola Data *Membership*

Deskripsi singkat	:	Admin menambah, mengubah, menghapus, dan melihat data *membership*.
Aktor	:	Admin
Pre kondisi	:	Admin terautentikasi.
Pos Kondisi	:	Perubahan tersimpan pada table *membership*
Basic Flow	:	Menampilkan data *membership*
Alternative Flow	:	Admin memilih *membership* → edit → simpan perubahan.
Eror Flow	:	Mengubah data tidak sesuai ketentuan

Identifikasi		UC-201
Nama Use Case	:	Kelola Data Produk
Deskripsi singkat	:	Admin menambah, mengubah, menghapus, dan melihat data produk.
Aktor	:	Admin
Pre kondisi	:	Admin terautentikasi.
Pos Kondisi	:	Perubahan tersimpan pada table produk
Basic Flow	:	Menampilkan data produk
Alternative Flow	:	Admin memilih produk → edit → simpan perubahan.
Eror Flow	:	Mengubah data tidak sesuai ketentuan

Identifikasi		UC-101
Nama Use Case	:	Kelola Data *Private Trainer*
Deskripsi singkat	:	Admin menambah, menghapus, dan melihat data *private trainer*.
Aktor	:	Admin

Pre kondisi	:	Admin terautentikasi.
Pos Kondisi	:	Perubahan tersimpan pada table *private trainer*
Basic Flow	:	Menampilkan data *private trainer*
Alternative Flow	:	Admin memilih *private trainer* → edit → simpan perubahan.
Eror Flow	:	Mengubah data tidak sesuai ketentuan

Identifikasi	:	UC-PM-001
Nama Use Case	:	Beli Paket *Membership*
Deskripsi singkat	:	*Member* memilih paket, melakukan checkout, mengunggah bukti pembayaran; status menunggu verifikasi.
Aktor	:	Pelanggan, Admin (verifikasi)
Pre kondisi	:	Pelanggan *login* dan paket tersedia.
Pos Kondisi	:	Transaksi tercatat dengan status “Menunggu Verifikasi”.
Basic Flow	:	1. Pelanggan memilih paket dan konfirmasi order. 2. Sistem membuat record transaksi dan menampilkan instruksi pembayaran. 3. Pelanggan unggah bukti (JPG/PNG/PDF ≤5MB). 4. Sistem menyimpan file dan ubah status transaksi menjadi “Menunggu Verifikasi”. 5. Admin menerima notifikasi verifikasi.
Alternative Flow	:	Jika membayar lewat cash → admin buat transaksi manual dan set verifikasi.
Eror Flow	:	File bukti invalid/terlalu besar → tolak upload. Nominal tidak sesuai → tandai transaksi perlu pemeriksaan.

Identifikasi	:	UC-PM-003
Nama Use	:	Verifikasi Pembayaran

Case		
Deskripsi singkat	:	Admin memeriksa bukti pembayaran dan menerima/menolak transaksi.
Aktor	:	Admin
Pre kondisi	:	Ada transaksi dengan status “Menunggu Verifikasi”.
Pos Kondisi	:	Jika diterima → status transaksi “Disetujui” dan *membership* aktif; jika ditolak → status “Ditolak” dan diberi keterangan.
Basic Flow	:	1. Admin membuka daftar transaksi menunggu verifikasi. 2. Memeriksa bukti dan detail transaksi. 3. Memilih setuju atau tolak. 4. Sistem memperbarui status transaksi dan mengirim notifikasi ke pelanggan.
Alternative Flow	:	Admin meminta bukti tambahan dari pelanggan
Eror Flow	:	Bukti tidak terbaca → admin meminta unggahan ulang

Identifikasi	:	UC-PP-001
Nama Use Case	:	Beli Produk & Update Stok
Deskripsi singkat	:	Pelanggan membeli produk; sistem mengurangi stok ketika transaksi disetujui.
Aktor	:	Admin, pelanggan
Pre kondisi	:	Produk tersedia dan stok > 0.
Pos Kondisi	:	Jika transaksi disetujui → stok dikurangi; jika ditolak → stok tetap.
Basic Flow	:	1. Pelanggan pilih produk dan jumlah. 2. Sistem cek stok, buat transaksi dan instruksi pembayaran. 3. Pelanggan unggah bukti. 4. Admin verifikasi → jika ok, sistem update stok dan set status transaksi Disetujui.

Alternative Flow	:	Jika stok tidak mencukupi → sistem menolak pembelian
Eror Flow	:	Barang tidak tersedia

Identifikasi	:	UC-801-
Nama Use Case	:	Buat Jadwal Latihan
Deskripsi singkat	:	Pelanggan menjadwalkan sesi latihan dengan *private trainer* dan memastikan tidak ada bentrok.
Aktor	:	Pelanggan, *Private Trainer* (*verifikasi*)
Pre kondisi	:	*Private Trainer* dan pelanggan terdaftar; pelanggan aktif.
Pos Kondisi	:	Jadwal tersimpan dan *Private Trainer* meverifikasi.
Basic Flow	:	1. Pelanggan dan *Private Trainer* membuka form jadwal. 2. Memilih tanggal, jam mulai & selesai. 3. Sistem melakukan cek bentrok. 4. Jika tidak bentrok dan simpan jadwal.
Alternative Flow	:	Jika bentrok → harus memilih jadwal lain
Eror Flow	:	Jam mulai ≥ jam selesai → tampilkan pesan kesalahan.

Identifikasi	:	UC-601-
Nama Use Case	:	Laporan
Deskripsi singkat	:	Admin menghasilkan laporan transaksi, laporan pelanggan aktif, dan laporan jadwal
Aktor	:	Admin
Pre kondisi	:	Admin *login*; data transaksi/kepelangganan tersedia.
Pos Kondisi	:	Laporan dihasilkan
Basic Flow	:	1. Admin pilih jenis laporan dan rentang tanggal.

		2. Sistem menjalankan query, menyiapkan laporan preview.
Alternative Flow	:	Menyimpan template laporan untuk penggunaan berkala.
Eror Flow	:	Query timeout pada rentang sangat besar → minta batasi rentang atau jalankan background job

Identifikasi	:	UC-801-02
Nama Use Case	:	Lihat Jadwal
Deskripsi singkat	:	pelanggan / *Private Trainer* melihat jadwal personal dalam tampilan list.
Aktor	:	pelanggan, *private trainer*
Pre kondisi	:	User *login*.
Pos Kondisi	:	Tampilkan jadwal sesesuai yang dibuat pengguna sebelumnya
Basic Flow	:	1. User buka menu jadwal. 2. Sistem menampilkan jadwal yang relevan (tanggal, jam).
Alternative Flow	:	Filter berdasarkan tanggal, atau status.
Eror Flow	:	Jika data jadwal tidak tersedia → tampilkan “Tidak ada jadwal”.

Identifikasi	:	UC-EP-
Nama Use Case	:	Edit Profil
Deskripsi singkat	:	Pengguna dapat memperbarui informasi profil dan *password*.
Aktor	:	Admin, pelanggan, *private trainer*
Pre kondisi	:	User *login*.
Pos Kondisi	:	Data profil diperbarui di DB.
Basic Flow	:	1. User buka halaman profil. 2. Memperbarui *field* yang diizinkan.

		3. Sistem memvalidasi dan menyimpan perubahan.
Alternative Flow	:	Ubah *password* memerlukan konfirmasi *password* lama.
Eror Flow	:	*Username* baru sudah dipakai → tolak perubahan.

3.4 Kebutuhan Non-Fungsional

3.4.1 Performance

ID Requirement	:	NFR-PF-001
Deskripsi	:	1. Halaman *dashboard* (Admin) harus merender dan dapat dipakai *end-to-end* dalam waktu ≤ 3 detik pada beban normal (hingga 50 concurrent users). 2. Halaman *input* (form registrasi, upload bukti, form jadwal) harus merespons aksi submit dalam ≤ 2 detik. 3. Query laporan harian (rentang 1 hari, hingga 10.000 record) harus selesai dan menampilkan hasil dalam ≤ 5 detik. 4. Operasi ringan (lihat detail pelanggan, lihat jadwal) ≤ 1,5 detik.
Rasional	:	Menjamin pengalaman pengguna yang responsif sehingga admin, *private trainer*, dan pelanggan dapat bekerja efisien dan manajemen memperoleh informasi tepat waktu

3.4.2 Reliability

ID Requirement	:	NFR-RL-001
Deskripsi	:	1. Semua operasi kritis (pencatatan transaksi *membership*, verifikasi pembayaran, update stok produk, pembuatan jadwal) harus dijalankan dalam konteks transaksi *database* (ACID)

		sehingga tidak terjadi kehilangan atau inkonsistensi data saat gagal. 2. Sistem harus tahan terhadap kegagalan parsial: apabila proses verifikasi gagal di tengah jalan, keadaan data harus kembali ke status konsisten *(rollback).* 3. Maksimum tingkat kehilangan data yang diizinkan akibat kegagalan sistem = 0% untuk transaksi yang sudah dikonfirmasi; untuk data sementara (*form* yang belum submit) toleransi = < 0,1%.
Rasional	:	Menjamin integritas data finansial dan kepelangganan agar laporan, tagihan, dan jadwal tetap akurat.

3.4.3 Availability

ID Requirement	:	NFR-AV-001
Deskripsi	:	1. Sistem harus tersedia minimal 99.5% *uptime* per bulan selama jam operasional utama (06:00–22:00). 2. Maintenance terjadwal harus dilakukan di luar jam operasional utama; pemberitahuan maintenance ke pengguna harus dikirim minimal 48 jam sebelumnya. 3. *Downtime* tak terduga maksimum yang diizinkan ≤ 0.5% per bulan.
Rasional	:	Ketersediaan tinggi dibutuhkan agar admin dan pelanggan dapat mengakses fungsi kritis (registrasi, verifikasi pembayaran, penjadwalan) tanpa gangguan.

3.4.4 Security

ID Requirement	:	NFR-SC-001
Deskripsi	:	Sistem menerapkan autentikasi pengguna, dan kontrol hak akses

		berbasis peran
Rasional	:	Melindungi data sensitif perusahaan dan mencegah akses tidak sah yang dapat merusak operasi atau menimbulkan kerugian.

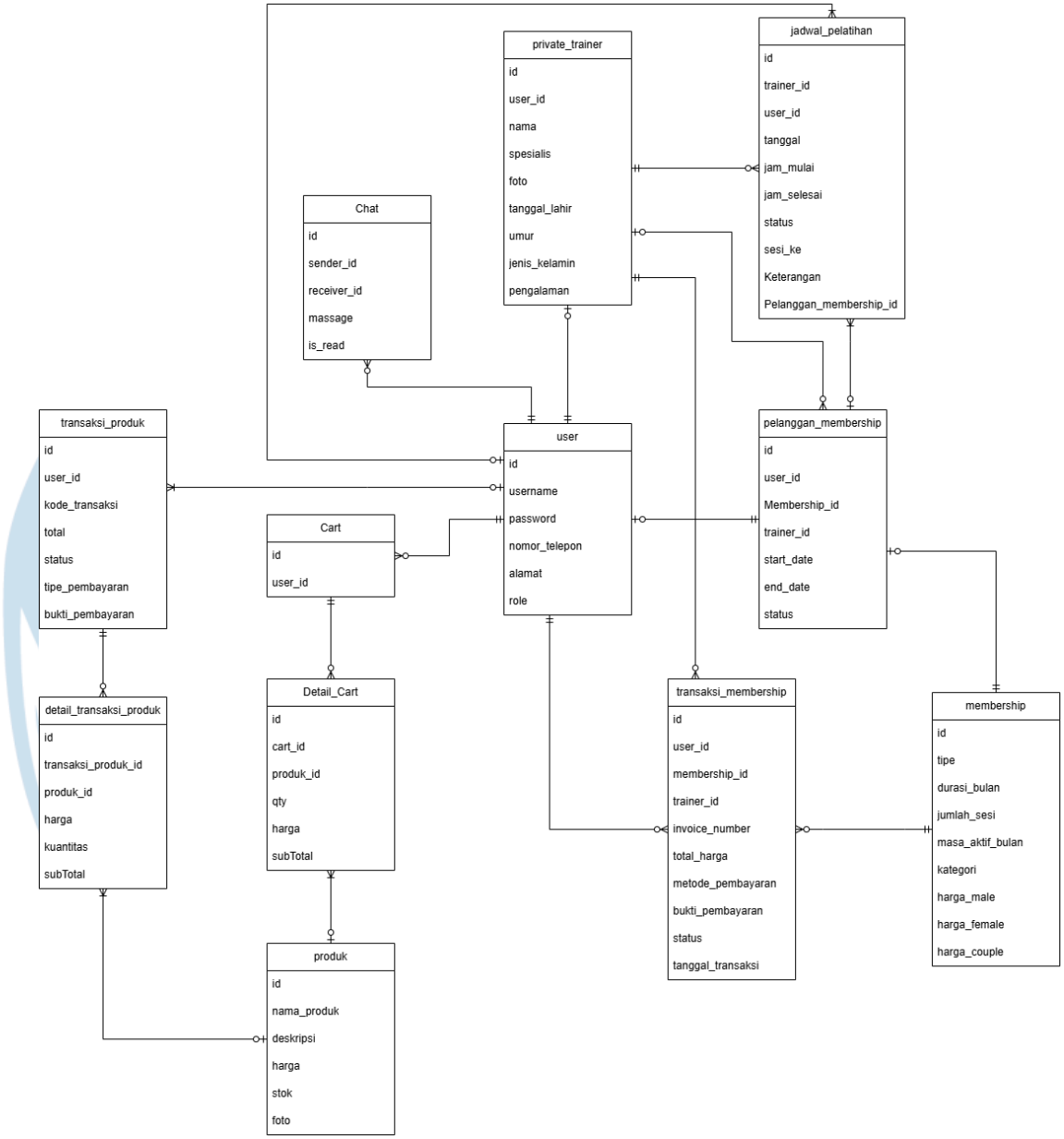
3.4.5 Maintainability

ID Requirement	:	NFR-MT-001
Deskripsi	:	Kode sumber disusun *modular* dan terdokumentasi, disimpan di sistem *version control*
Rasional	:	Mempermudah perbaikan dan pengembangan fitur baru sehingga sistem tetap relevan dan stabil dalam jangka panjang.

3.5.6 Usability

ID Requirement	:	NFR-US-001
Deskripsi	:	Antarmuka mudah dipahami oleh pengguna operasional
Rasional	:	Memastikan adopsi sistem oleh pengguna lapangan sehingga manfaat efisiensi dan akurasi dapat tercapai.

3.5 Kebutuhan Data



PEMBANGUNAN SISTEM INFORMASI *GYM*
BERBASIS *WEB*SITE MENGGUNAKAN

Deskripsi Perancangan Perangkat Lunak

Axel Christian Nugroho
210711114

Dibuat untuk:
Tugas Akhir
2026

Daftar Revisi

Tanggal	Deskripsi Revisi	Penulis	Keterangan
<date>	<Version 1>	<Your Name>	<First Revision>

Persetujuan Dokumen

Dokumen Deskripsi Perancangan Perangkat Lunak ini telah diterima dan disetujui oleh:

Nama	Tanggal	Tandatangan

Daftar Isi

Daftar Revisi 143

Persetujuan Dokumen 143

1. Pendahuluan 145

1.1 Tujuan Penulisan Dokumen 145

1.2 Ruang Lingkup 146

1.3 Definisi, Akronim, dan Singkatan 149

1.4 Referensi 151

2. Perancangan Sistem 152

2.1 Perancangan Arsitektur 152

2.1.1. Overview Sistem 152

2.1.2. Arsitektur Perangkat Lunak 153

2.2 Perancangan Rinci 154

2.3 Perancangan Data 162

2.1.3. Dekomposisi Data 162

2.1.4. Physical Data Model 170

2.4 Perancangan AntarMuka 170

On Click ‘Reschedule’ 206

1. Pendahuluan

Dokumen Perancangan Perangkat Lunak (DPPL) merupakan dokumen yang berisi informasi yang dibutuhkan oleh pengembang perangkat lunak untuk merancang produk perangkat lunak Sistem Informasi GYM Berbasis *Web*site. Dokumen ini berisi rincian mengenai bagaimana perangkat lunak akan dikembangkan, yang meliputi arsitektur perangkat lunak, dekomposisi data, perancangan basis data, deskripsi antarmuka, perancangan modul-modul sistem, serta informasi teknis pendukung lainnya.

DPPL ini berfungsi sebagai pedoman bagi pengembang selama proses implementasi dan memastikan bahwa perangkat lunak yang dibangun sesuai dengan Spesifikasi Kebutuhan Perangkat Lunak (SKPL) serta dapat memenuhi kebutuhan pengguna secara fungsional maupun non-fungsional. Dokumen ini juga membantu memastikan konsistensi desain, kemudahan pemeliharaan, dan keberlanjutan pengembangan sistem di masa mendatang.

Sistem Informasi GYM ini dirancang untuk mendukung proses operasional gym, termasuk pengelolaan data pelanggan, pengelolaan *private trainer*, paket pelangganship, transaksi pembayaran, penjadwalan latihan, dan pembuatan laporan. Dengan adanya DPPL, seluruh komponen perangkat lunak dapat dirancang secara terstruktur, modular, dan mudah dikembangkan sesuai kebutuhan organisasi.

1.1 Tujuan Penulisan Dokumen

Tujuan penulisan Dokumen Perancangan Perangkat Lunak (DPPL) ini adalah untuk menyediakan acuan teknis yang lengkap bagi pihak-pihak yang terlibat dalam proses pengembangan Sistem Informasi *GYM* Berbasis *Web*site. Dokumen ini disusun untuk:

1. Pelangganikan panduan bagi pengembang perangkat lunak, sehingga setiap komponen sistem dapat dirancang dan diimplementasikan sesuai dengan kebutuhan yang telah didefinisikan dalam Spesifikasi Kebutuhan Perangkat Lunak (SKPL).

2. Menjadi referensi bagi tim penguji dalam memahami struktur sistem, alur logika, serta modul-modul yang akan diuji, sehingga proses pengujian dapat dilakukan secara sistematis dan menyeluruh.
3. Menjadi dokumentasi teknis bagi pihak pemelihara sistem agar proses perawatan, pembaruan, dan pengembangan lanjutan dapat dilakukan secara efisien dan terstruktur.
4. Pelangganikan gambaran menyeluruh kepada pemangku kepentingan (*stakeholder*) mengenai rancangan arsitektur, desain modul, alur data, dan mekanisme kerja sistem yang akan dibangun.

Dengan adanya DPPL ini, seluruh pihak yang berperan dalam proses pengembangan diharapkan memiliki pemahaman yang sama mengenai desain sistem, sehingga implementasi dapat dilakukan secara konsisten, terarah, dan sesuai dengan standar yang ditetapkan.

1.2 Ruang Lingkup

Subbagian ini menjelaskan cakupan perancangan dan pengembangan perangkat lunak Sistem Informasi *GYM* Berbasis *Web*site. Ruang lingkup mencakup identifikasi produk perangkat lunak, batasan fungsi yang akan dan tidak akan dikembangkan, serta manfaat dan sasaran yang ingin dicapai oleh sistem.

1. Produk Perangkat Lunak yang Akan Dihasilkan

Pengembangan sistem ini akan menghasilkan beberapa komponen perangkat lunak sebagai berikut:

1. Aplikasi *Web* Sistem Informasi *GYM*
    a. Dibangun menggunakan *framework* Laravel (PHP).
    b. Menggunakan MySQL sebagai *Database Management System* (DBMS).
    c. Menggunakan *Blade Template Engine* untuk tampilan antarmuka.
2. Modul Pengelolaan Data (CRUD Module)
    a. Modul data pelanggan
    b. Modul data *private trainer*
    c. Modul paket pelanggan*ship*

    d. Modul produk (suplemen/*merchandise*)
3. Modul Transaksi
    a. Pembelian pelanggan*ship*
    b. Pembelian produk
    c. *Upload* dan verifikasi bukti pembayaran
4. Modul Penjadwalan Latihan
    a. Pembuatan jadwal latihan
    b. Deteksi bentrokan jadwal otomatis
    c. Tampilan jadwal untuk pelanggan dan *private trainer*
5. Modul Laporan (*Report Generator*)
    a. Laporan transaksi pelanggan*ship*
    b. Laporan produk
    c. Laporan keaktifan pelanggan
    d. Laporan jadwal dan aktivitas *private trainer*
6. Modul Keamanan
    a. Autentikasi dan otorisasi berbasis *role*
    b. Proteksi *CSRF* dan validasi *input*
7. *Dashboard* Admin
    a. Ringkasan data operasional *gym*
    b. Pemantauan transaksi dan aktivitas pelanggan

2. Apa yang Akan dan Tidak Akan Dilakukan oleh Sistem

A. Sistem Akan Melakukan:

a. Mendukung pengelolaan data pelanggan*ship* dan *private trainer* secara lengkap.
b. Memproses transaksi pembelian pelanggan*ship* dan produk berbasis *upload* bukti pembayaran.
c. Mengelola jadwal latihan antara pelanggan dan *private trainer*.
d. Menyediakan laporan operasional *gym* yang dapat *diekspor* (PDF/CSV).
e. Menyediakan autentikasi dan hak akses berbasis *role*.
f. Menyediakan *dashboard* untuk admin sebagai pusat pengelolaan.

B. Sistem Tidak Akan Melakukan:

a. Tidak menyediakan *payment gateway* otomatis (Midtrans, Xendit, dsb.).
b. Tidak menyediakan aplikasi *mobile native* (Android/iOS).
c. Tidak mengintegrasikan *IoT* atau sensor fisik *gym*.
d. Tidak mengelola fitur absensi berbasis kartu/pengenalan wajah.
e. Tidak menangani manajemen alat *gym* (inventaris peralatan).

Batasan ini selaras dengan ruang lingkup yang telah dibahas dalam laporan TA.

3. Manfaat, Sasaran, dan Tujuan Penerapan Aplikasi

Pengembangan Sistem Informasi *GYM* ini bertujuan untuk pelangganikan manfaat sebagai berikut:

A. Manfaat Sistem

a. Meningkatkan efisiensi operasional *gym*, menggantikan pencatatan manual yang memakan waktu dan rawan kesalahan.
b. Mempermudah proses administrasi, termasuk verifikasi pembayaran dan manajemen *membership*.
c. Memfasilitasi penjadwalan latihan, meminimalkan bentrok jadwal antara pelanggan dan *private trainer*.
d. Pelangganikan akses informasi yang cepat kepada pelanggan mengenai status *membership*, transaksi, dan jadwal latihan.
e. Memperbaiki kualitas layanan melalui sistem yang terstruktur dan modern.

B. Tujuan Sistem

a. Menghadirkan sistem informasi terintegrasi yang mendukung pengelolaan *membership*, jadwal latihan, pembayaran, serta laporan operasional.
b. Mengurangi human eror dan meningkatkan akurasi data.
c. Pelangganikan pengalaman pengguna yang lebih baik bagi admin, pelanggan, dan *private trainer*.
d. Menjadi landasan pengembangan sistem lanjutan jika *gym* ingin memperluas layanan digital di masa depan.

1.3 Definisi, Akronim, dan Singkatan

Definisi	
Admin	Pengguna internal yang memiliki akses penuh untuk mengelola data pelanggan, *private trainer*, paket *membership*, transaksi pembayaran, jadwal latihan, dan laporan operasional *gym*.
Private trainer	Pelatih pribadi yang pelangganikan layanan latihan khusus kepada pelanggan. *Private trainer* memiliki jadwal kerja dan dapat mengelola serta melihat jadwal latihan di sistem.
Pelanggan	Pengguna yang terdaftar dan memiliki akun untuk membeli paket *membership*, latihan dengan *private trainer*, dan mengakses fitur sistem lainnya.
Dashboard	Halaman utama pengguna setelah *login* yang menampilkan ringkasan informasi seperti jumlah pelanggan aktif, transaksi terbaru, jadwal latihan, dan status pembayaran.
Jadwal Latihan	Informasi mengenai waktu latihan yang dijadwalkan antara pelanggan dan *private trainer*, meliputi tanggal, jam mulai, dan jam selesai
Pelanggan *Membership*	Status kepelangganan pelanggan yang ditentukan berdasarkan paket yang dibeli, dengan tanggal mulai dan tanggal berakhir masa aktifnya.
Paket *membership*	Jenis layanan kepelangganan yang ditawarkan oleh *gym*, misalnya paket 1 bulan, 3 bulan, 6 bulan atau 1 tahun, termasuk harga, durasi.
Pembayaran	Proses transaksi yang dilakukan pelanggan untuk membeli paket *membership* atau produk *gym*, yang harus diverifikasi oleh admin melalui bukti pembayaran.
Produk *Gym*	Barang yang dijual oleh *gym* seperti suplemen.
Laporan	Dokumen atau tampilan digital yang memuat informasi operasional *gym* seperti laporan transaksi, laporan keaktifan pelanggan, laporan jadwal latihan, dan laporan produk.
Riwayat Transaksi	Catatan seluruh transaksi pembelian *membership* atau produk yang dilakukan oleh pelanggan.

Definisi	
Admin	Pengguna internal yang memiliki akses penuh untuk mengelola data pelanggan, *private trainer*, paket *membership*, transaksi pembayaran, jadwal latihan, dan laporan operasional *gym*.
Private trainer	Pelatih pribadi yang pelangganikan layanan latihan khusus kepada pelanggan. *Private trainer* memiliki jadwal kerja dan dapat mengelola serta melihat jadwal latihan di sistem.
Pelanggan	Pengguna yang terdaftar dan memiliki akun untuk membeli paket *membership*, latihan dengan *private trainer*, dan mengakses fitur sistem lainnya.
Dashboard	Halaman utama pengguna setelah *login* yang menampilkan ringkasan informasi seperti jumlah pelanggan aktif, transaksi terbaru, jadwal latihan, dan status pembayaran.
Jadwal Latihan	Informasi mengenai waktu latihan yang dijadwalkan antara pelanggan dan *private trainer*, meliputi tanggal, jam mulai, dan jam selesai
Stok Produk	Jumlah persediaan produk *gym* yang tersedia untuk dijual.
Transaksi	Pencatatan pembelian *membership* atau produk oleh pelanggan, berikut statusnya (menunggu verifikasi, disetujui, ditolak).
Upload Bukti Pembayaran	Fitur di mana pelanggan mengunggah bukti pembayaran berupa gambar atau PDF untuk diverifikasi admin.

Akronim dan Singkatan	
CRUD	*Create, Read, Update, Delete* – Operasi dasar manajemen data pada sistem.
DBMS	*Database Management System* – Perangkat lunak manajemen basis data yang digunakan, yaitu MySQL.
DPPL	*Dokumen Perancangan Perangkat Lunak* – Dokumen teknis yang menjelaskan perancangan sistem.
ERD	*Entity Relationship Diagram* – Diagram hubungan antar entitas data.
GYM	Pusat kebugaran tempat pelanggan melakukan aktivitas latihan.

Akronim dan Singkatan	
CRUD	*Create, Read, Update, Delete* – Operasi dasar manajemen data pada sistem.
DBMS	*Database Management System* – Perangkat lunak manajemen basis data yang digunakan, yaitu MySQL.
DPPL	*Dokumen Perancangan Perangkat Lunak* – Dokumen teknis yang menjelaskan perancangan sistem.
HTML	*HyperText Markup Language* – Bahasa *markup* untuk tampilan antarmuka pengguna.
HTTP / HTTPS	Protokol komunikasi *web*; HTTPS adalah versi aman menggunakan *enkripsi* SSL/TLS.
MVC	*Model-View-Controller* – Arsitektur yang digunakan oleh Laravel.
MySQL	DBMS relasional yang dipakai untuk menyimpan data sistem *gym*.
RBAC	*Role-Based Access Control* – Mekanisme kontrol hak akses sistem.
SUS	*System Usability Scale* – Metode evaluasi *usability* sistem.

1.4 Referensi

Laporan Tugas Akhir ”PEMBANGUNAN SISTEM INFORMASI GYM BERBASIS *WEB*SITE”.

2. Perancangan Sistem

2.1 Perancangan Arsitektur

Subbab ini menggambarkan bagaimana Sistem Informasi *GYM* diperinci menjadi komponen-komponen penyusun, arsitektur logis aplikasi, dan tata letak *deployment* (penyebaran).

2.1.1. Overview Sistem

Tujuan: pelanggani gambaran konteks dan *boundary* sistem.

Sistem Informasi *GYM* adalah aplikasi *web* berbasis Laravel yang melayani tiga aktor utama: Admin, Pelanggan, dan *Private trainer*. Sistem menyediakan modul: autentikasi, manajemen pelanggan, manajemen *private trainer*, paket *membership*, transaksi (*upload* bukti), verifikasi, penjadwalan latihan, dan modul laporan. Sistem dirancang untuk berjalan pada arsitektur client–*server* dengan antarmuka *web* responsif.

Konteks sistem (high-level):

a. Klien: *browser* pada *desktop* / *mobile* (*responsive web*).
b. *Server* Aplikasi: instance Laravel (PHP-FPM) di belakang *web server* (Apache).
c. *Database*: MySQL untuk penyimpanan utama.
d. *Storage*: filesystem untuk *upload* bukti pembayaran (lokal atau object *storage* S3).
e. Monitoring & *Backup*: *service* untuk *uptime*, log aggregation, dan *backup* DB.

Boundary:

a. Sistem menerima *upload* bukti pembayaran dari *user*, tetapi tidak terintegrasi dengan *payment gateway*.
b. Fitur *mobile native* / *IoT* / absensi *hardware*.

2.1.2. Arsitektur Perangkat Lunak

Sistem menggunakan arsitektur n-layer (3-layer / MVC) yang dipetakan ke struktur Laravel:

Lapisan (*logical layers*):

1. *Presentation Layer (View)*

   Blade templates / frontend assets (HTML/CSS/JS).

   Tugas: render UI, validasi sisi-klien, menampilkan notifikasi.

2. *Application Layer* (*Controller* & *Service*)

   Laravel *Controllers* menerima *request*, memanggil *Service*s / Use-case classes.

   *Service*s / Use-case: meng-enkapsulasi logika bisnis (mis. proses verifikasi pembayaran, cek bentrok jadwal).

   Form Request classes untuk validasi *input*.

3. *Domain* / *Data Layer* (*Model & Repository*)

   Eloquent Models (*User*, Pelanggan, *Private Trainer*, Package, Transaction, Schedule, Product).

   Repositories / Query Builder untuk akses DB, memastikan *single responsibility* dan *testability*.

4. Infrastructure Layer
   a. *Database* (MySQL)
   b. *Storage* (local / S3)
   c. *Cache/Queue* (Redis)
   d. Mail (SMTP)
   e. Logging / *Monitoring*

Komponen utama / *package dependency*:

a. laravel/*framework* (*core*)
b. laravel/ui atau laravel/jetstream
c. spatie/laravel-permission (RBAC)
d. intervention/*image* (*image* processing untuk *upload* bukti)
e. laravel/telescope (*debugging*, dev)

Package / modul aplikasi (logical modules):

a. Auth Module (*login*, *register*, *role*s)
b. Pelanggan *Management Module*
c. *Private Trainer Management Module*
d. *Package & Product Module*
e. *Transaction Module* (*upload*, status, audit)
f. *Scheduling Module* (create, conflict detection)
g. *Reporting Module*
h. *Dashboard Module*

2.2 Perancangan Rinci

Deskripsi atribut kelas *User*

Nama atribut	Tipe Data	Deskripsi
id	Big int	*Primary key* identitas *user*
username	Varchar	Nama pengguna untuk *login*
password	Varchar	Kata sandi *user*
nomor_telepon	Varchar	Nomor telepon *user*
alamat	Text	Alamat tempat tinggal pengguna
role	varchar	Peran *user* (admin, pelanggan, *private trainer*)

Deskripsi fungsi *login* pada kelas *User*

Login (*username*, *password*)	
Input	*username*, *password*
Output	status autentikasi
Deskripsi	Melakukan proses *login* pengguna.

Deskripsi fungsi *register* pada kelas *User*

register(data*User*)	
Input	Data *user*
Output	Data *user* baru tersimpan
Deskripsi	Mendaftarkan *user* baru ke sistem.

Deskripsi atribut kelas *Private trainer*

Nama Atribut	Tipe Data	Deskripsi
id	BigInt	*Primary key private trainer*
*user*_id	BigInt	*Foreign key* ke tabel *User*
nama	Varchar	Nama *private trainer*
spesialis	Varchar	Deskripsi spesialis *private trainer*
foto	Varchar	Tampilan foto *private trainer*
tanggal_lahir	date	Informasi tanggal lahir *private trainer*
umur	Int	Informasi umur *private trainer*
jenis_kelamin	varchar	Informasi jenis kelamin *private trainer*
pengalaman	Text	Deskripsi pengalaman singkat *private trainer*

Deskripsi fungsi updateBiodata pada kelas *Private trainer*

*update*Profil(data *private Trainer*)	
Input	Data *private Trainer* baru yang ingin diubah
Output	Data *private trainer* terbarui
Deskripsi	Memperbarui biodata *private trainer*

Deskripsi atribut kelas *Membership*

Nama Atribut	Tipe Data	Deskripsi
id	BigInt	*Primary key membership*
tipe	Varchar	Jenis *membership*
Durasi_bulan	int	Durasi lama *membership*
Jumlah_sesi	int	Total sesi latihan dengan *private trainer*
Masa_aktif_bulan	int	Berapa lama *membership* aktif
ketegori	varchar	Jenis kategori *membership*
harga_male	Decimal	Harga untuk pria
harga_female	Decimal	Harga untuk wanita
harga_couple	Decimal	Harga untuk dua orang / pasangan

Deskripsi fungsi *update*Pelanggan*ship* pada kelas *membership*

getHarga(tipe)	
Input	Pilih data yang ingin diubah
Output	Data terbaru tersimpan pada *database*
Deskripsi	Memperbarui data yang ingin diubah

Deskripsi atribut kelas Pelanggan_ *membership*

Nama Atribut	Tipe Data	Deskripsi
id	BigInt	*Primary key*
user _id	BigInt	*foreign key* ke pelanggan
membership _id	BigInt	*Foreign key* ke *membership*

*Trainer*_id	BingInt	*foreign key* ke *private trainer*
start_date	Date	Tanggal mulai *membership*
end_date	Date	Tanggal berakhir *membership*
status	varchar	Status keaktifan *membership*

Deskripsi fungsi cekStatus pada kelas Pelanggan_Pelanggan*ship*

cekStatus()	
Input	tanggal saat ini
Output	Data status aktif/nonaktif
Deskripsi	Mengecek status kepelangganan *user*

Deskripsi atribut kelas Jadwal_Pelatihan

Nama Atribut	Tipe Data	Deskripsi
id	BigInt	*Primary key* jadwal
*trainer*_id	BigInt	*Foreign key* ke *Private_Trainer*
Pelanggan*Membership*_id	BigInt	*Foreign key* ke pelanggan *Membership*
Nama_pelatihan	Varchar	Nama sesi latihan
tanggal	Date	Tanggal latihan
Jam_mulai	Time	Waktu mulai
Jam_selesai	Time	Waktu selesai
status	varchar	Status pengajuan

		jadwal Latihan
Pelanggan_id	BigInt	*Foreign key* ke pelanggan *User*
Sesi_ke	Int	Pelanggani informasi latihan sesi dengan *private trainer*
keterangan	text	Informasi tambahan pada jawal pelatihan

Deskripsi fungsi buatJadwal pada kelas Jadwal_Pelatihan

buatJadwal(dataJadwal)	
Input	Data Jadwal
Output	Data jadwal baru tersimpan
Deskripsi	Membuat jadwal pelatihan.

Deskripsi atribut kelas Produk

Nama Atribut	**Tipe Data**	**Deskripsi**
id	BigInt	*Primary key* produk
nama_produk	Varchar	Nama produk
deskripsi	Text	Deskripsi produk
harga	Decimal	Harga produk
stok	Int	Jumlah stok produk
foto	Varchar	Menampilkan visual produk

Deskripsi fungsi *update*Stok pada kelas Produk

*update*Stok(jumlah)	
Input	Masukan jumlah barang yang ingin ditambahkan pada produk tertentu
Output	Stok produk terbaru
Deskripsi	Memperbarui stok produk

Deskripsi atribut kelas Transaksi_ *membership*

Nama Atribut	Tipe Data	Deskripsi
id	BigInt	*Primary key* order
*user*_id	BigInt	*Foreign key* ke *User*
Pelanggan*ship*_id	BigInt	*Foreign key* ke *membership*
*Trainer*_id	BigInt	*Foreign key* ke *User private trainer*
Invoice_number	Varchar	Kode pembayaran
Total_harga	BigInt	Total pembayaran
Metode_pembayaran	varchar	Metode yang digunakan dalam transaksi
Status	Varchar	Status order
Bukti_pembayaran	Varchar	*Upload* bukti pembayaran
Tanggal_transaksi	date	Tanggal transaksi yang dilaksanakan

Deskripsi fungsi buatOrder pada kelas Transaksi_ *membership*

store(Order)	
Input	Membuat order transaksi *membership* baru
Output	Data order transaksi baru tersimpan
Deskripsi	Membuat transaksi pembelian *membership*

Deskripsi atribut kelas *Chat*

Nama Atribut	Tipe Data	Deskripsi
id	BigInt	*Primary key* order
sender_id	BigInt	*Foreign key* ke

		pengirim
receiver_id	BigInt	*Foreign key* ke Penerima
Message	text	Pesan yang ingin dikirim

Deskripsi fungsi buatChat pada kelas *Chat*

kirimPesan(pesan)	
Input	Memasukan pesan yang ingi ditulis
Output	Pesan tampil pada penerima
Deskripsi	Percakapan pesan antar pelanggan dan *private trainer*

Deskripsi atribut kelas *Cart*

Nama Atribut	Tipe Data	Deskripsi
id	BigInt	*Primary key cart*
*user*_id	BigInt	*Foreign key* ke *User*

Deskripsi atribut Kelas *Detail Cart*

Nama Atribut	Tipe Data	Deskripsi
id	BigInt	*Primary key detail cart*
*Cart*_id	BigInt	*Foreign key* ke *cart*
Produk_id	BigInt	*Foreign key* ke produk
Qty	Int	Jumlah produk dalam *cart*
Harga	Decimal	Harga setiap produk
subtotal	Decimal	Total harga dalam *cart*

Deskripsi fungsi buat*Cart* pada kelas *Cart*

buat*Cart*(keranjang)	
Input	Memasukan produk yang ingin dibeli
Output	Data tersimpan pada *Cart*

Deskripsi	Menyimpan data produk yang akan dibeli pada *Cart*

Deskripsi atribut kelas Transaksi_Produk

Nama Atribut	Tipe Data	Deskripsi
id	BigInt	*Primary key* order
*user*_id	BigInt	*Foreign key* ke *User*
Kode_transaksi	Varchar	Kode pemesanan
Total	BigInt	Total pembayaran
status	Varchar	Status order
Tipe_pembayaran	Varchar	Tipe pembayaran yang dipakai
Bukti_pembayaran	Varchar	*Upload* bukti pembayaran

Deskripsi fungsi buatOrder pada kelas Transaksi_Produk

kirimPesan(pesan)	
Input	Data Order baru
Output	Data order baru tersimpan
Deskripsi	Membuat transaksi pembelian.

Deskripsi atribut kelas Detail_Transaksi_Produk

Nama Atribut	Tipe Data	Deskripsi
id	BigInt	*Primary key*
Transaksi_produk_id	BigInt	*Foreign key* ke Order
produk_id	BigInt	*Foreign key* ke Produk
harga	BigInt	Harga prduk
kuantitas	int	Jumlah produk yang dibeli
subtotal	BigInt	Total harga item

Deskripsi fungsi hitungSubtotal pada kelas Detail_Transaksi_Produk

kirimPesan(pesan)	
Input	Masukkan harga dan kuantitas
Output	subtotal
Deskripsi	Menghitung total harga setelah dihitung dengan banyaknya barang

2.3 Perancangan Data

2.1.3. Dekomposisi Data

Tabel 1 : Struktur Tabel *User*

Nama Field	Tipe Data	Null	Constraint	Default	Keterangan	Nama Field
id	BIGINT	No	PK, AI	–	Identitas *user*	id
username	VARCHAR(100)	No	UNIQUE	–	Nama *login*	*username*
password	VARCHAR(255)	No	–	–	*Password* terenkripsi	*password*
nomor_telepon	VARCHAR(20)	No	–	–	Nomor telepon	nomor_telepon
alamat	TEXT	Yes	–	NULL	Alamat *user*	alamat
role	VARCHAR(20)	No	–	pelanggan	*Role user*	*role*

Tabel 2 : Struktur Tabel *Private_Trainer*

Nama Field	Tipe Data	Null	Constraint	Keterangan	Nama Field	Tipe Data
id	BIGINT	No	PK, AI	ID *private trainer*	id	BIGINT
*user*_id	BIGINT	No	FK	Relasi ke	*user*_id	BIGINT

				user		
nama	VARCHAR	No	–	Nama *private trainer*	nama	VARCHAR
spesialis	VARCHAR	No	–	Spesialis	spesialis	VARCHAR
foto	VARCHAR	Yes	–	Foto	foto	VARCHAR
tanggal_lahir	DATE	No	–	TTL	tanggal_lahir	DATE
umur	INT	No	–	Umur	umur	INT
jenis_kelamin	VARCHAR	No	–	Gender	jenis_kelamin	VARCHAR
pengalaman	TEXT	Yes	–	Pengalaman	pengalaman	TEXT

Tabel 3 : Struktur Tabel *membership*

Nama Field	**Tipe Data**	**Null**	**Constraint**	**Keterangan**	**Nama Field**	**Tipe Data**
id	BIGINT	No	PK, AI	ID *membership*	id	BIGINT
tipe	VARCHAR	No	–	Jenis paket	tipe	VARCHAR
durasi_bulan	INT	No	–	Lama *membership*	durasi_bulan	INT
jumlah_sesi	INT	No	–	Total sesi	jumlah_sesi	INT
masa_aktif_bulan	INT	No	–	Masa aktif	masa_aktif_bulan	INT
kategori	VARCH	No	–	Kategori	kategori	VARCH

	AR					AR
harga_male	DECIMAL	No	–	Harga pria	harga_male	DECIMAL
harga_female	DECIMAL	No	–	Harga wanita	harga_female	DECIMAL
harga_couple	DECIMAL	No	–	Harga pasangan	harga_couple	DECIMAL

Tabel 4 : Struktur Tabel Pelanggan_ *membership*

Nama Field	Tipe Data	Null	Constraint	Keterangan	Nama Field	Tipe Data
id	BIGINT	No	PK, AI	ID	id	BIGINT
*user*_id	BIGINT	No	FK	Relasi ke *user*	*user*_id	BIGINT
membership _id	BIGINT	No	FK	Relasi *membership*	*membership* _id	BIGINT
*trainer*_id	BIGINT	Yes	FK	*private Trainer* opsional	*trainer*_id	BIGINT
start_date	DATE	No	–	Tanggal mulai	start_date	DATE
end_date	DATE	No	–	Tanggal berakhir	end_date	DATE
status	VARCHAR	No	–	Status aktif	status	VARCHAR

Tabel 5 : Struktur Tabel Jadwal_Pelatihan

Nama Field	Tipe Data	Null	Constraint	Keterangan	Nama Field	Tipe Data
id	BIGINT	No	PK, AI	ID jadwal	id	BIGINT

*trainer*_id	BIGINT	No	FK	*private Trainer*	*trainer*_id	BIGINT
pelanggan_id	BIGINT	NO	FK	*User*	pelanggan_id	BIGINT
pelanggan*Membership*_id	BIGINT	NO	FK	Pelanggan *Membership*	Pelanggan_*membership*_id	BIGINT
nama_pelatihan	VARCHAR	No	–	Nama sesi	nama_pelatihan	VARCHAR
tanggal	DATE	No	–	Tanggal	tanggal	DATE
jam_mulai	TIME	No	–	Jam mulai	jam_mulai	TIME
jam_selesai	TIME	No	–	Jam selesai	jam_selesai	TIME
sesi_ke	INT	No	–	Sesi ke	sesi_ke	INT
status	VARCHAR	No	–	Status	status	VARCHAR
keterangan	TEXT	Yes	–	Catatan	keterangan	TEXT

Tabel 6 : Struktur Tabel Produk

Nama Field	**Tipe Data**	**Null**	**Constraint**	**Keterangan**	**Nama Field**	**Tipe Data**
Id	BIGINT	No	PK, AI	ID produk	id	BIGINT
nama_produk	VARCHAR	No	–	Nama	nama_produk	VARCHAR
deskripsi	TEXT	Yes	–	Deskripsi	deskripsi	TEXT
harga	DECIMAL	No	–	Harga	harga	DECIMAL
Stok	INT	No	–	Stok	stok	INT

Foto	VARCHAR	Yes	–	Gambar	foto	VARCHAR

Tabel 7 : Struktur Tabel *Cart*

Nama Field	Tipe Data	Null	Constraint	Keterangan	Nama Field	Tipe Data
Id	BIGINT	No	PK, AI	ID	id	BIGINT
*user*_id	BIGINT	No	FK	*User*	*user*_id	BIGINT

Tabel 8 : Struktur Tabel Detail_*Cart*

Nama Field	Tipe Data	Null	Constraint	Keterangan	Nama Field	Tipe Data
Id	BIGINT	No	PK, AI	ID	id	BIGINT
*cart*_id	BIGINT	No	FK	*Cart*	*cart*_id	BIGINT
produk_id	BIGINT	No	FK	Produk	produk_id	BIGINT
Qty	INT	No	–	Jumlah	kuantitas	INT
harga	DECIMAL	NO	-	Harga	harga	DECIMAL
subtotal	DECIMAL	No	–	Total	subtotal	DECIMAL

Tabel 9 : Struktur Tabel Chat

Nama Field	Tipe Data	Null	Constraint	Keterangan	Nama Field	Tipe Data
Id	BIGINT	No	PK, AI	ID	id	BIGINT
sender_id	BIGINT	No	FK	Pengirim	*sender_id*	BIGINT
receiver_i	BIGIN	No	FK	Penerima	*receiver_i*	BIGIN

d	T				*d*	T
message	TEXT	No	–	Isi pesan	*message*	TEXT

Tabel 10 : Struktur Tabel Transaksi_ *membership*

Nama Field	**Tipe Data**	**Null**	**Constraint**	**Keterangan**	**Nama Field**	**Tipe Data**
Id	BIGINT	No	PK, AI	ID	id	BIGINT
*user*_id	BIGINT	No	FK	*User*	*user*_id	BIGINT
*membership*_id	BIGINT	No	FK	*membership*	*membership*_id	BIGINT
*trainer*_id	BIGINT	Yes	FK	*private Trainer*	*trainer*_id	BIGINT
invoice_number	VARCHAR	No	UNIQUE	Invoice	invoice_number	VARCHAR
Total_harga	INT	NO	-	Total harga	Total_harga	INT
Metode_pembayaran	VARCHAR	NO	-	Metode pembayaran yang dipakai	Metode_pembayaran	VARCHAR
Bukti_pembayaran	VARCHAR	NO	-	Bukti pembayaran transaksi	Bukti_pembayaran	VARCHAR
Status	VARCHAR	NO	-	Status transaks	Status	VARCHAR

				i		
Tanggal_transaksi	TIMESTAMP	NO	-	Tanggal traksasi	Tanggal_transaksi	TIMESTAMP

Tabel 11 : Struktur Tabel Transaksi_Produk

Nama Field	Tipe Data	Null	Constraint	Keterangan	Nama Field	Tipe Data
Id	BIGINT	No	PK, AI	ID	id	BIGINT
*user*_id	BIGINT	No	FK	*User*	*user*_id	BIGINT
kode_transaksi	VARCHAR	No	UNIQUE	Kode	kode_transaksi	VARCHAR
Total	BIGINT	No	–	Total	total	BIGINT
Status	VARCHAR	No	–	Status	status	VARCHAR
tipe_pembayaran	VARCHAR	No	–	Metode	tipe_pembayaran	VARCHAR
bukti_pembayaran	VARCHAR	Yes	–	Bukti	bukti_pembayaran	VARCHAR

Tabel 12 : Struktur Tabel Detail_Transaksi_Produk

Nama Field	Tipe Data	Null	Constraint	Keterangan	Nama Field	Tipe Data
Id	BIGINT	No	PK, AI	ID	id	BIGINT
transaksi_produk_id	BIGINT	No	FK	Transaksi	transaksi_produk_id	BIGINT
produk_id	BIGINT	No	FK	Produk	produk_id	BIGINT
harga	BIGINT	No	–	Harga	harga	BIGINT

kuantitas	INT	No	–	Jumlah	kuantitas	INT
subtotal	BIGINT	No	–	Total	subtotal	BIGINT

Nama Field : nama atribut dalam tabel

Tipe Data : tipe data atribut

Null : apakah diperbolehkan bernilai *Null* atau tidak. *Yes* berarti boleh bernilai null, *No* tidak boleh bernilai null.

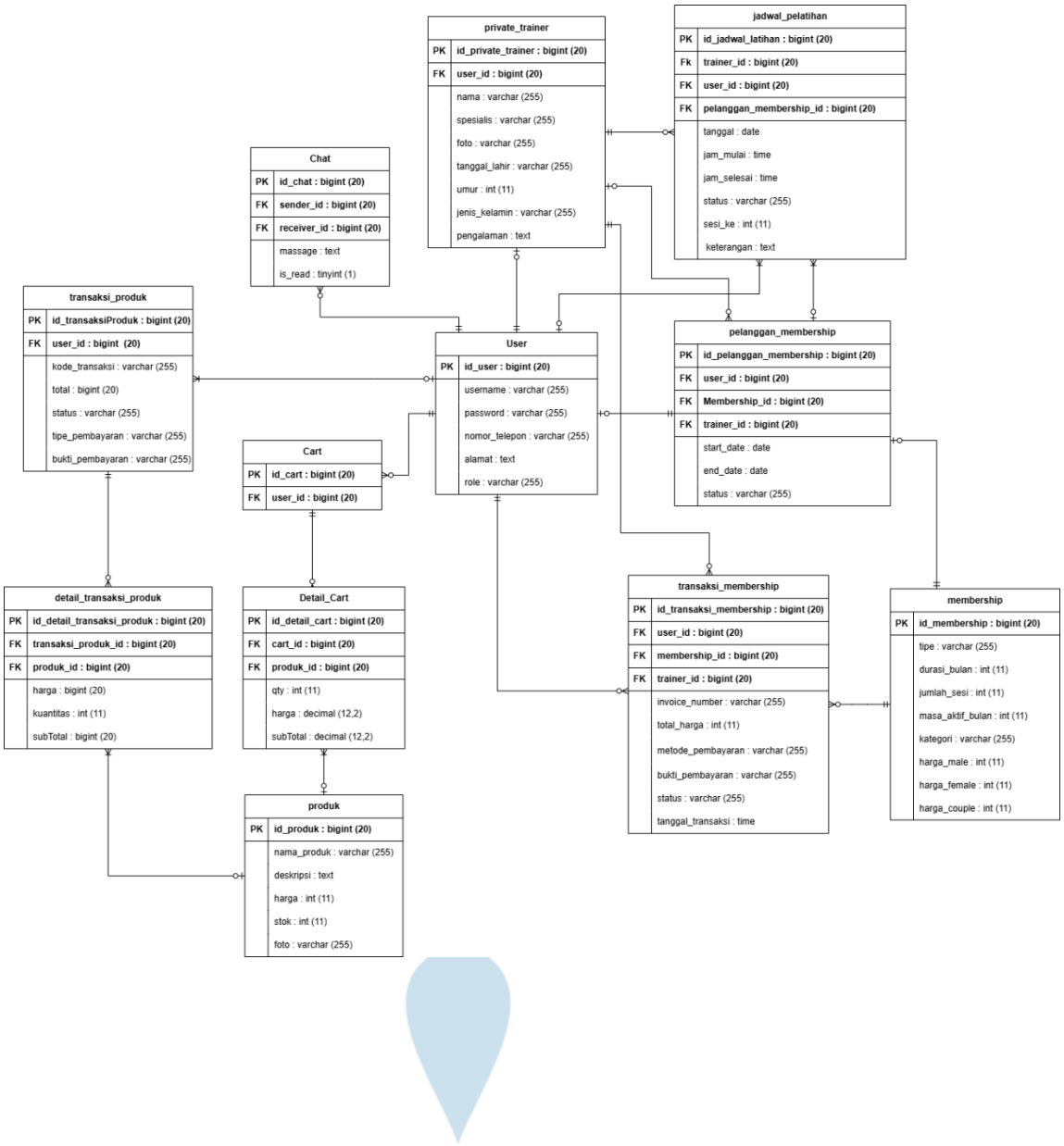
Default : nilai *default* atribut

Keterangan : deskripsi tambahan (jika diperlukan) dari atribut.

Konstrain : konstrain yang dimiliki atribut

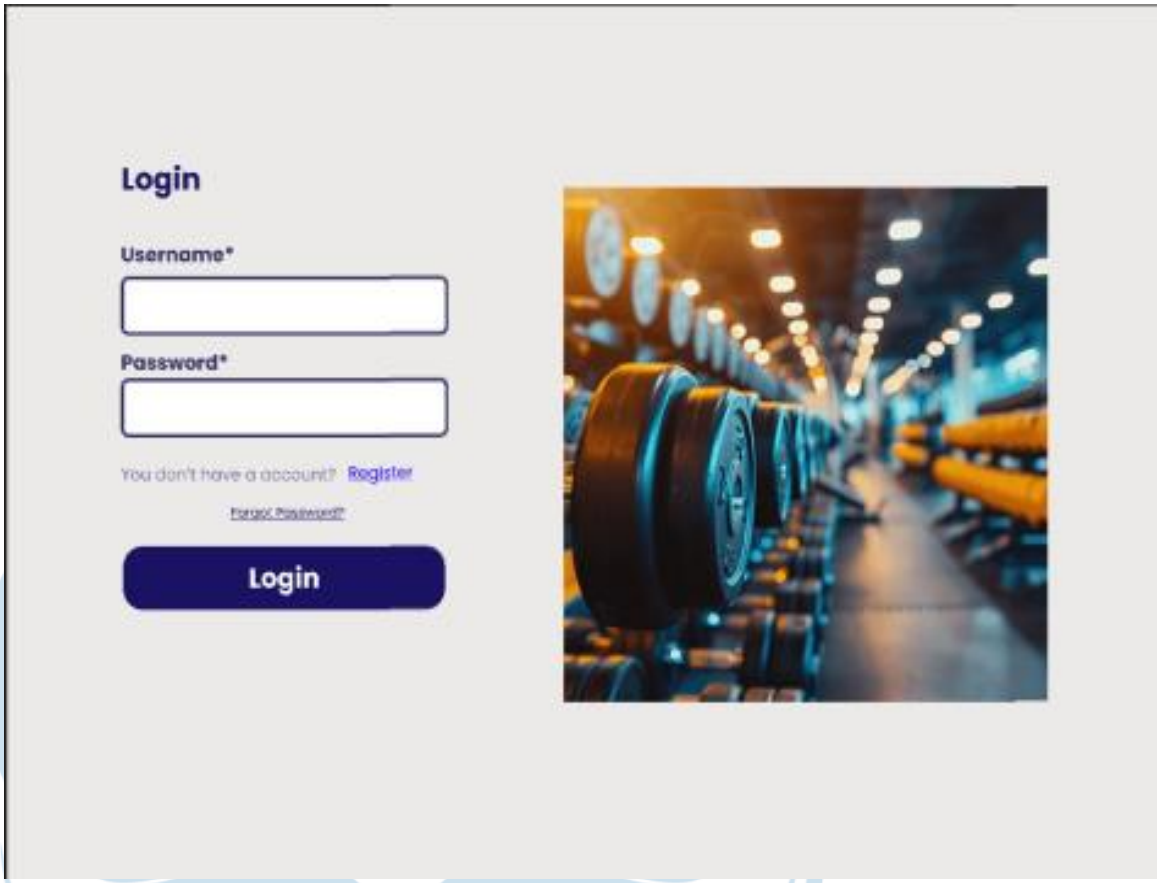
Range Nilai : jangkauan atau enumerasi nilai yang *valid* (diijinkan)

2.1.4. Physical Data Model



2.4 Perancangan Antarmuka

1. Antarmuka Kelola *Login*



Deskripsi : Antarmuka Login digunakan oleh pengguna untuk melakukan autentikasi sebelum dapat mengakses sistem informasi. Pengguna diwajibkan memasukkan username dan password yang valid sesuai data yang tersimpan pada basis data. Antarmuka ini menyediakan fasilitas validasi input, pesan kesalahan, serta navigasi ke halaman pendaftaran akun dan pemulihan kata sandi.

Input : Username dan Password

Output : Pesan berhasil login dan sistem menampilkan halaman utama sesuai hak

akses pengguna

Pesan peringatan jika username atau password tidak valid

Method/Algoritma : On ‘Login’

1. Baca data input username dan password.
2. Lakukan validasi data input.
3. Jika ada data yang kosong, tampilkan pesan peringatan “Username dan Password harus diisi”, kemudian kembali ke langkah 1.
4. Lakukan proses autentikasi dengan mencocokkan username dan password ke basis data pada tabel User.
5. Jika data tidak ditemukan atau tidak sesuai, tampilkan pesan “Username atau Password salah” dan kembali ke langkah 1.
6. Jika data valid, sistem menampilkan pesan “Login berhasil”.
7. Sistem mengarahkan pengguna ke halaman utama sesuai dengan hak akses.

On ‘Form Login’

1. Menampilkan form login.
2. Mengosongkan field username dan password.

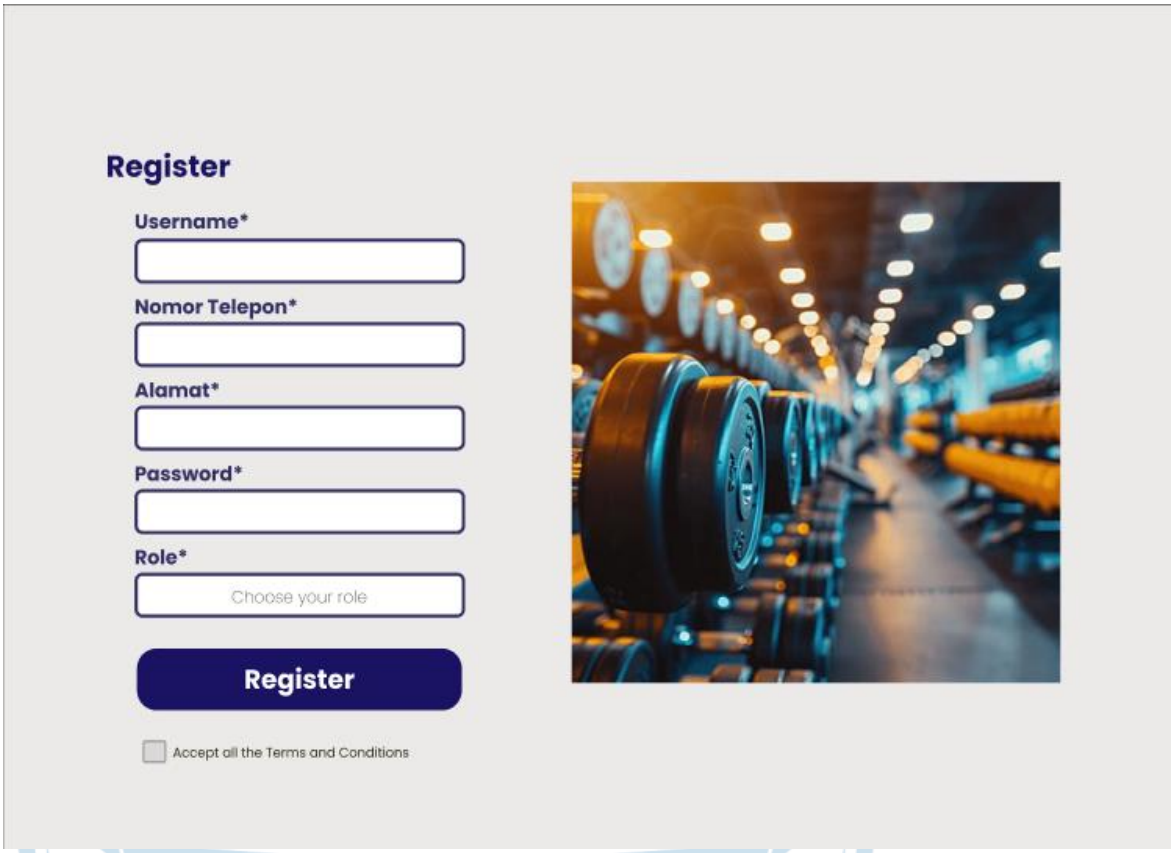
On ‘Click Register’

1. Sistem mengarahkan pengguna ke halaman pendaftaran akun.

On Load ‘Forgot Password’

1. Sistem mengarahkan pengguna ke halaman pemulihan kata sandi.

2.Antarmuka *Register*



Deskripsi : Antarmuka Register digunakan oleh pengguna untuk melakukan pendaftaran akun baru ke dalam sistem informasi. Pengguna diwajibkan mengisi data identitas berupa username, nomor telepon, alamat, password, dan role. Sistem akan melakukan validasi data sebelum menyimpan data pengguna ke dalam basis data.

Input : Username, Nomor Telepon, Alamat, Password, Role

Output : Pesan pendaftaran berhasil
Akun pengguna tersimpan di basis

data

Pengguna diarahkan ke halaman Login

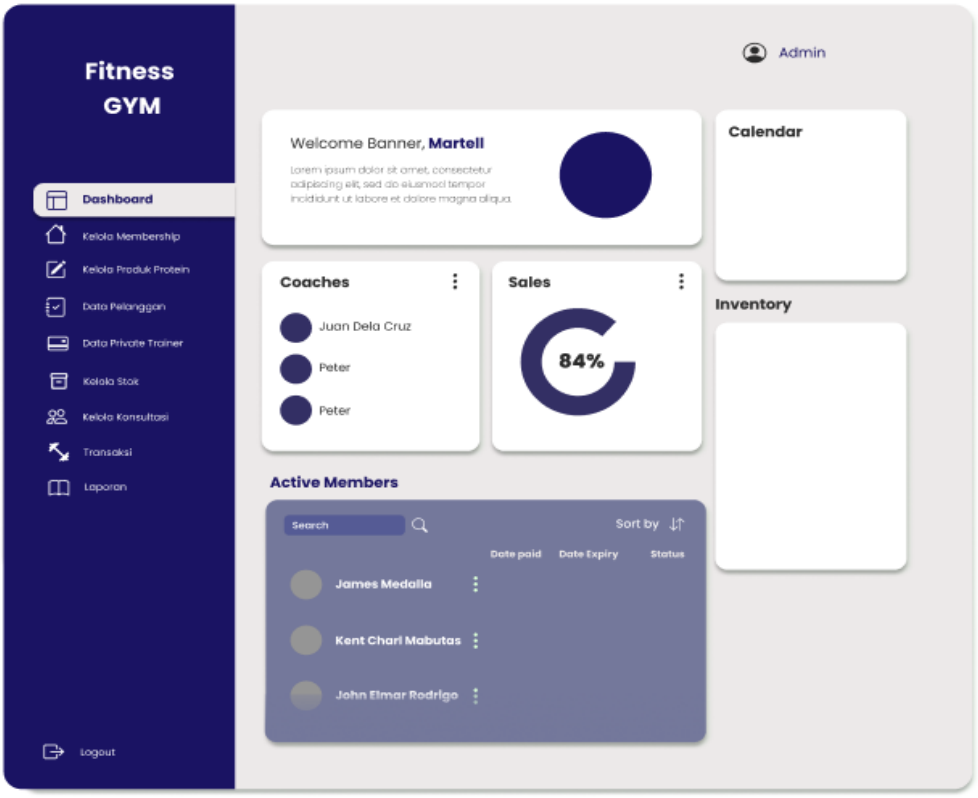
Method/Algoritma : On ‘Register’

1. Baca seluruh data input (username, nomor telepon, alamat, password, role).
2. Lakukan validasi data input.
3. Jika terdapat data yang kosong atau tidak valid, tampilkan pesan peringatan “Ada data yang belum diisi atau tidak valid”, kemudian kembali ke langkah 1.
4. Periksa apakah username sudah terdaftar di basis data.
5. Jika username sudah digunakan, tampilkan pesan “Username sudah terdaftar”, kemudian kembali ke langkah 1.
6. Jika data valid, simpan data pengguna ke dalam basis data pada tabel User.
7. Menampilkan pesan “Registrasi berhasil”.
8. Sistem mengarahkan pengguna ke halaman Login.

On ‘Form Register’

1. Menampilkan form registrasi.
2. Mengosongkan seluruh field input.

3.Antarmuka *Dashboard* Admin



Deskripsi : Antarmuka Dashboard digunakan untuk menampilkan ringkasan informasi dan statistik utama sistem informasi Gym secara real-time. Dashboard membantu administrator dalam memantau aktivitas sistem, seperti jumlah pelanggan aktif, data penjualan, stok, serta laporan penjualan bulanan dalam bentuk grafik dan tabel.

Input : Periode waktu (bulan / tahun)
Kata kunci pencarian data (opsional)

Opsi pengurutan data

Output : Informasi jumlah seluruh pelanggan
Informasi jumlah pelanggan aktif
Informasi total penjualan
Grafik persentase penjualan
Tabel penjualan per bulan
Tabel daftar pelanggan aktif

Method/Algoritma : On Load 'Dashboard'

1. Sistem mengambil data pengguna yang sedang login.
2. Sistem menampilkan pesan selamat datang sesuai nama pengguna.
3. Sistem mengambil data jumlah seluruh pelanggan dari tabel Pelanggan.
4. Sistem mengambil data jumlah pelanggan aktif.
5. Sistem mengambil data total penjualan dari tabel Transaksi.
6. Sistem menampilkan ringkasan data dalam bentuk kartu informasi, grafik, dan tabel.

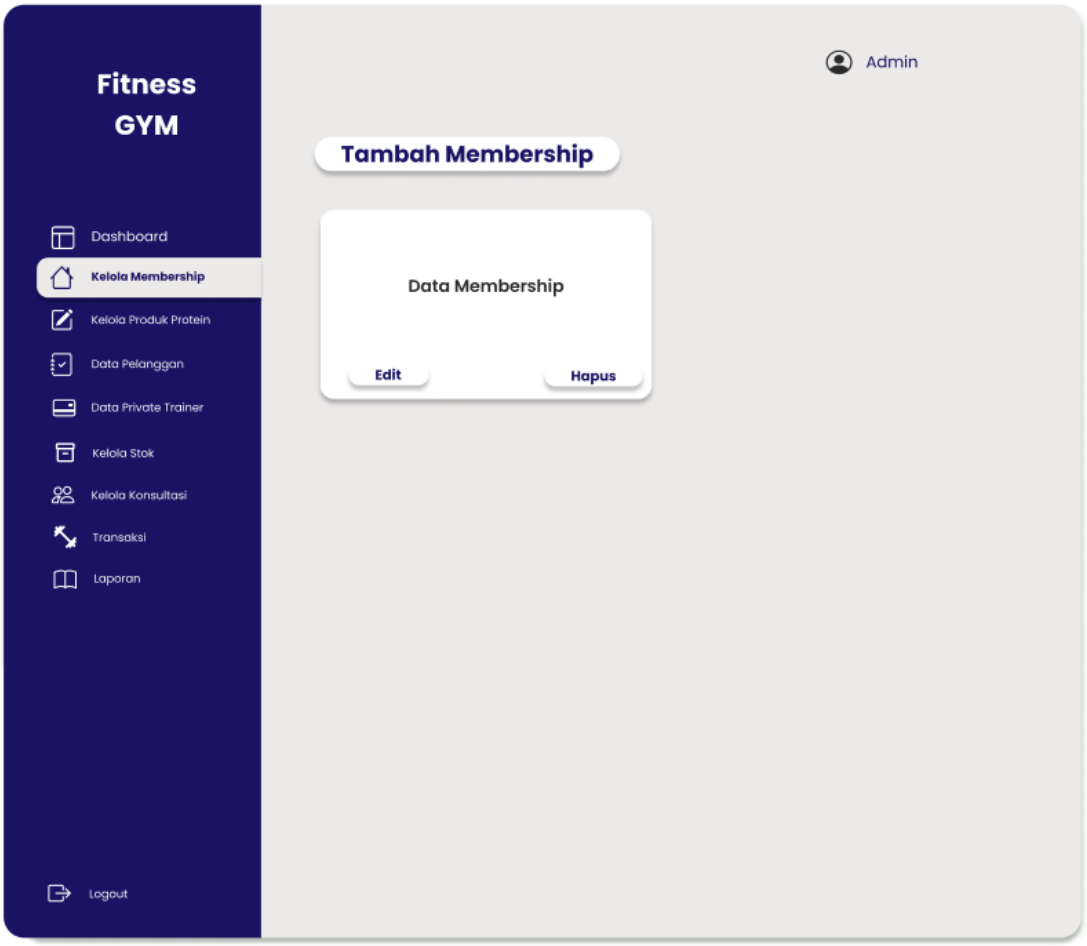
On Select 'Periode Waktu'

1. Sistem membaca periode waktu yang dipilih (bulan/tahun).
2. Sistem memfilter data penjualan berdasarkan periode tersebut.
3. Sistem memperbarui grafik dan tabel penjualan bulanan.

On Click ‘Logout’

1. Sistem mengakhiri sesi pengguna.
2. Sistem mengarahkan pengguna ke halaman Login

4.Antarmuka Admin Kelola *Membership*



Deskripsi	:	Antarmuka Kelola *Membership* digunakan oleh administrator untuk mengelola data paket *membership* gym yang tersedia dalam sistem. Administrator dapat melakukan proses tambah, ubah, hapus, dan melihat data *membership* yang digunakan oleh pelanggan. Antarmuka ini bertujuan untuk memudahkan pengelolaan paket *membership* secara terstruktur dan terintegrasi dengan data transaksi serta data pelanggan.
Input	:	Tipe *membership*, Harga *membership*
Output	:	Data *membership* ditampilkan dalam daftar / card data Pesan keberhasilan atau kegagalan proses pengelolaan data
Method/Algoritma	:	On Load 'Kelola Merbership' *1.* Sistem menampilkan halaman kelola *membership*. *2.* Sistem mengambil seluruh data *membership* dari tabel *Membership*. *3.* Sistem menampilkan data *membership* dalam bentuk daftar atau card. On Click 'Tambah *Membership*' *1.* Sistem menampilkan form tambah *membership*.

2. Administrator mengisi seluruh data *membership*.
3. Sistem membaca seluruh data input.
4. Sistem melakukan validasi data input.
5. Jika terdapat data tidak valid, tampilkan pesan "Ada data yang belum diisi atau tidak valid", kembali ke langkah 2.
6. Jika data valid, sistem menyimpan data *membership* ke tabel *membership*.
7. Sistem menampilkan pesan "Data *membership* berhasil ditambahkan".
8. Sistem memperbarui tampilan daftar *membership*.

On Click 'Edit'

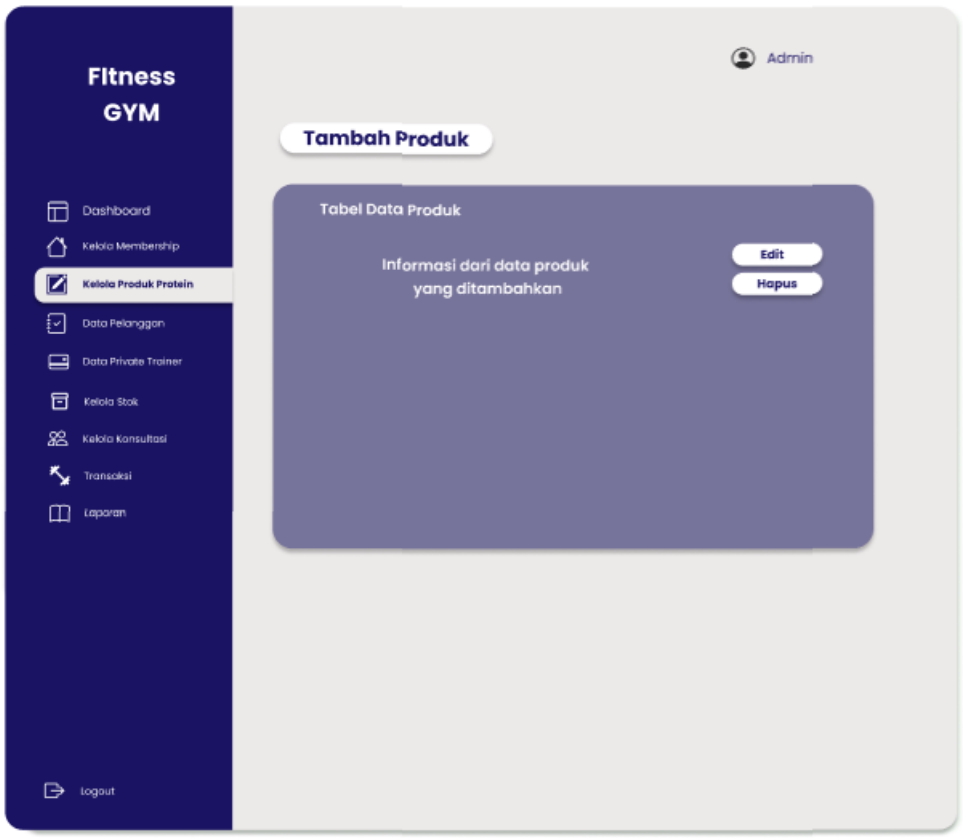
1. Sistem menampilkan form ubah data *membership* sesuai data yang dipilih.
2. Administrator mengubah data *membership*.
3. Sistem melakukan validasi data input.
4. Jika data valid, sistem memperbarui data *membership* di tabel *membership*.
5. Sistem menampilkan pesan "Data *membership* berhasil diperbarui".

6. Sistem memperbarui tampilan daftar *membership*.

On Click ’Hapus’

1. Sistem menampilkan konfirmasi penghapusan data.
2. Jika administrator menyetujui, sistem menghapus data *membership* dari tabel *membership*.
3. Sistem menampilkan pesan “Data *membership* berhasil dihapus”.
4. Sistem memperbarui tampilan daftar *membership*.

5.Antarmuka Admin Kelola Produk



Deskripsi	:	Antarmuka Kelola Produk Protein digunakan oleh administrator untuk mengelola data produk protein yang dijual di Gym. Administrator dapat melakukan proses tambah, ubah, hapus, dan melihat data produk protein. Antarmuka ini mendukung pengelolaan stok dan harga produk yang terintegrasi dengan modul transaksi dan laporan.
Input	:	Nama Produk, Deskripsi, Harga, Stok
Output	:	Data produk tersimpan di basis data Data produk ditampilkan dalam tabel data produk Pesan keberhasilan atau kegagalan proses pengelolaan data
Method/Algoritma	:	On Load 'Kelola Produk' *1.* Sistem menampilkan halaman kelola produk protein. *2.* Sistem mengambil seluruh data produk dari tabel Produk. *3.* Sistem menampilkan data produk dalam bentuk tabel data produk. On Click 'Tambah Produk' *1.* Sistem menampilkan form tambah produk protein.

2. Admin mengisi seluruh data produk.
3. Sistem membaca seluruh data input.
4. Sistem melakukan validasi data input.
5. Jika terdapat data tidak valid, tampilkan pesan “Ada data yang belum diisi atau tidak valid”, kembali ke langkah 2.
6. Jika data valid, sistem menyimpan data produk ke tabel Produk.
7. Sistem menampilkan pesan “Data produk berhasil ditambahkan”.
8. Sistem memperbarui tampilan tabel data produk.

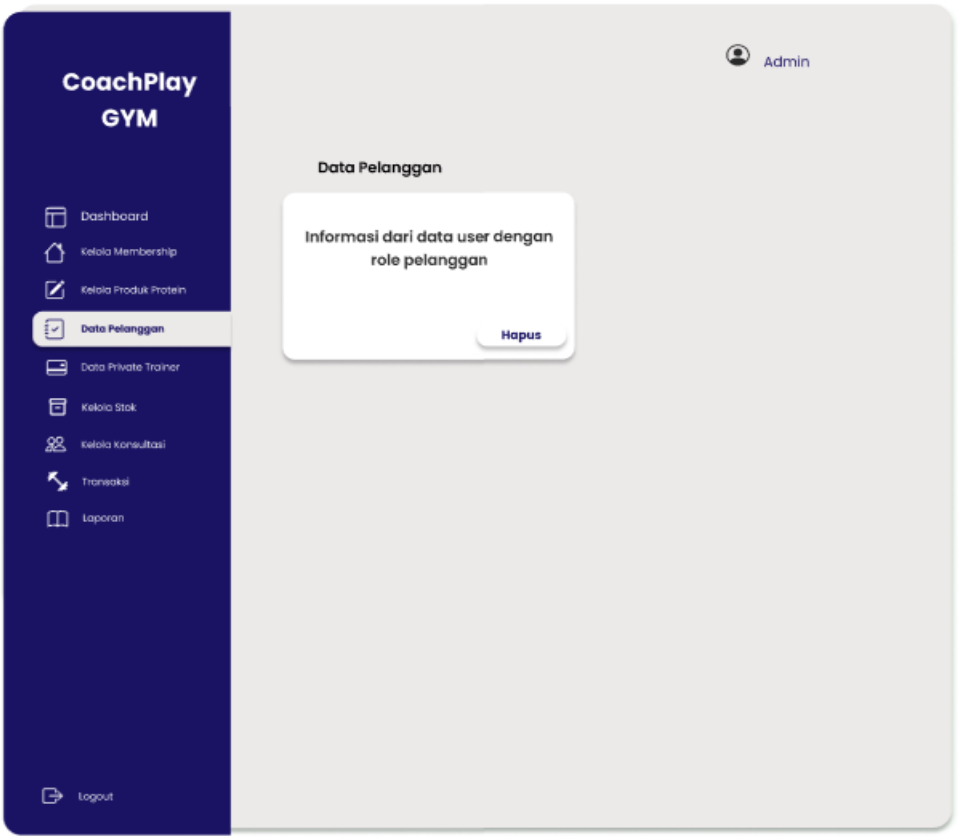
On Click ‘Edit’

1. Sistem menampilkan form ubah data produk sesuai data yang dipilih.
2. Administrator mengubah data produk.
3. Sistem melakukan validasi data input.
4. Jika data valid, sistem memperbarui data produk di tabel Produk.
5. Sistem menampilkan pesan “Data produk berhasil diperbarui”.
6. Sistem memperbarui tampilan tabel data produk.

On Click ’Hapus’

1. Sistem menampilkan konfirmasi penghapusan data.
2. Jika administrator menyetujui, sistem menghapus data produk dari tabel Produk.
3. Sistem menampilkan pesan “Data produk berhasil dihapus”.
4. Sistem memperbarui tampilan tabel data produk.

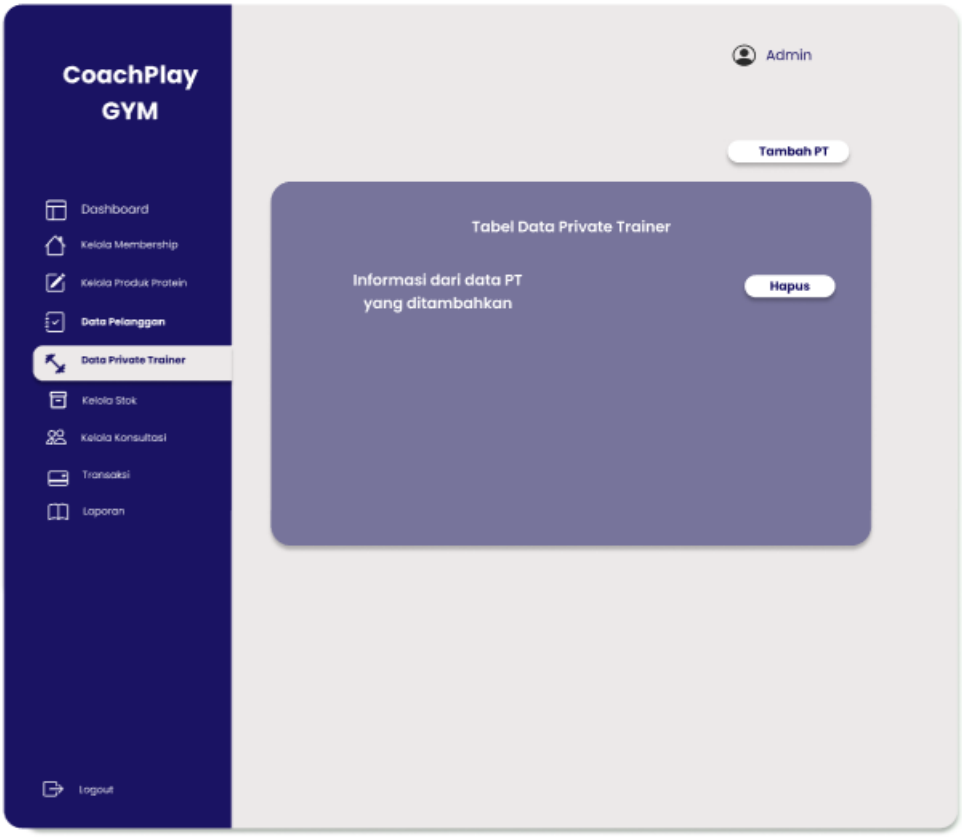
6.Antarmuka Admin Data Pelanggan



Deskripsi	:	Antarmuka Data Pelanggan digunakan oleh administrator untuk melihat dan mengelola data pelanggan yang terdaftar dalam sistem. Administrator hanya memiliki kewenangan untuk melihat data pelanggan dan menghapus data pelanggan tertentu. Penambahan dan perubahan data pelanggan dilakukan melalui proses registrasi dan transaksi.
Input	:	Kata kunci pencarian data pelanggan
Output	:	Data pelanggan ditampilkan dalam tabel data pelanggan Pesan konfirmasi dan pesan hasil penghapusan data pelanggan
Method/Algoritma	:	On Load 'Data Pelanggan' 1. Sistem menampilkan halaman data pelanggan. 2. Sistem mengambil seluruh data pelanggan dari tabel Pelanggan. 3. Sistem menampilkan data pelanggan dalam bentuk tabel data pelanggan. On Click 'Hapus' *1*. Sistem menampilkan konfirmasi penghapusan data pelanggan.

2. Jika administrator menyetujui, sistem menghapus data pelanggan dari tabel Pelanggan.
3. Sistem menampilkan pesan “Data pelanggan berhasil dihapus”.
4. Sistem memperbarui tampilan tabel data pelanggan.

7.Antarmuka Admin Data *Private trainer*



Deskripsi	:	Antarmuka Data *Private trainer* digunakan oleh administrator untuk menambah, melihat, mengelola data *private trainer* yang terdaftar dalam sistem. Administrator memiliki kewenangan untuk membuat, melihat data *private trainer* dan menghapus data *private trainer* tertentu.
Input	:	Kata kunci pencarian data *private trainer*
Output	:	Data *private trainer* ditampilkan dalam tabel data *private trainer*

Pesan konfirmasi dan pesan hasil penghapusan data *private trainer*

Method/Algoritma : On Load 'Data Memeber'

1. Sistem menampilkan halaman data *private trainer*.
2. Sistem mengambil seluruh data *private trainer* dari tabel *PrivateTrainer*.
3. Sistem menampilkan data *private trainer* dalam bentuk tabel data *private trainer*.

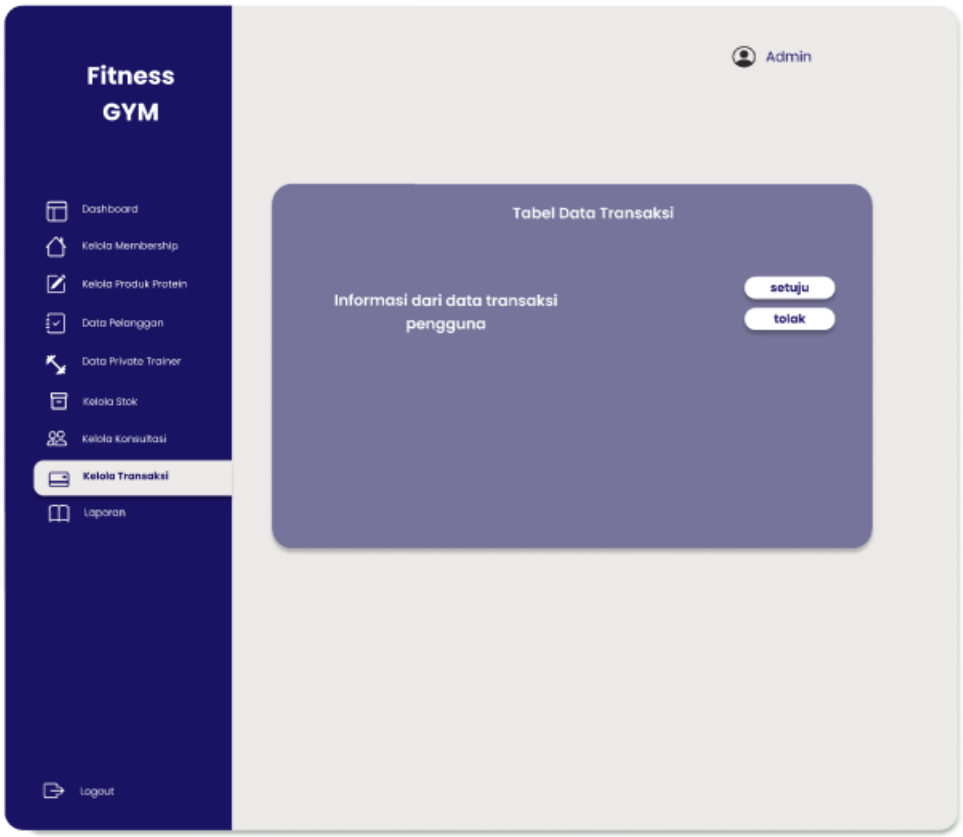
On Click 'Tambah*PrivateTrainer*'

1. Sistem menampilkan form tambah *private trainer*.
2. Administrator mengisi seluruh data user untuk *private trainer*.
3. Sistem membaca seluruh data input.
4. Sistem melakukan validasi data input.
5. Jika terdapat data tidak valid, tampilkan pesan "Ada data yang belum diisi atau tidak valid", kembali ke langkah 2.
6. Jika data valid, sistem menyimpan data *membership* ke tabel *membership*.
7. Sistem menampilkan pesan "Data *membership* berhasil ditambahkan".
8. Sistem memperbarui tampilan daftar *private trainer*.

On Click ’Hapus’

1. Sistem menampilkan konfirmasi penghapusan data *private trainer*.
2. Jika administrator menyetujui, sistem menghapus data *private trainer* dari tabel *PrivateTrainer*.
3. Sistem menampilkan pesan “Data *private trainer* berhasil dihapus”.
4. Sistem memperbarui tampilan tabel data *private trainer*.

8.Antarmuka Admin Data Transaksi

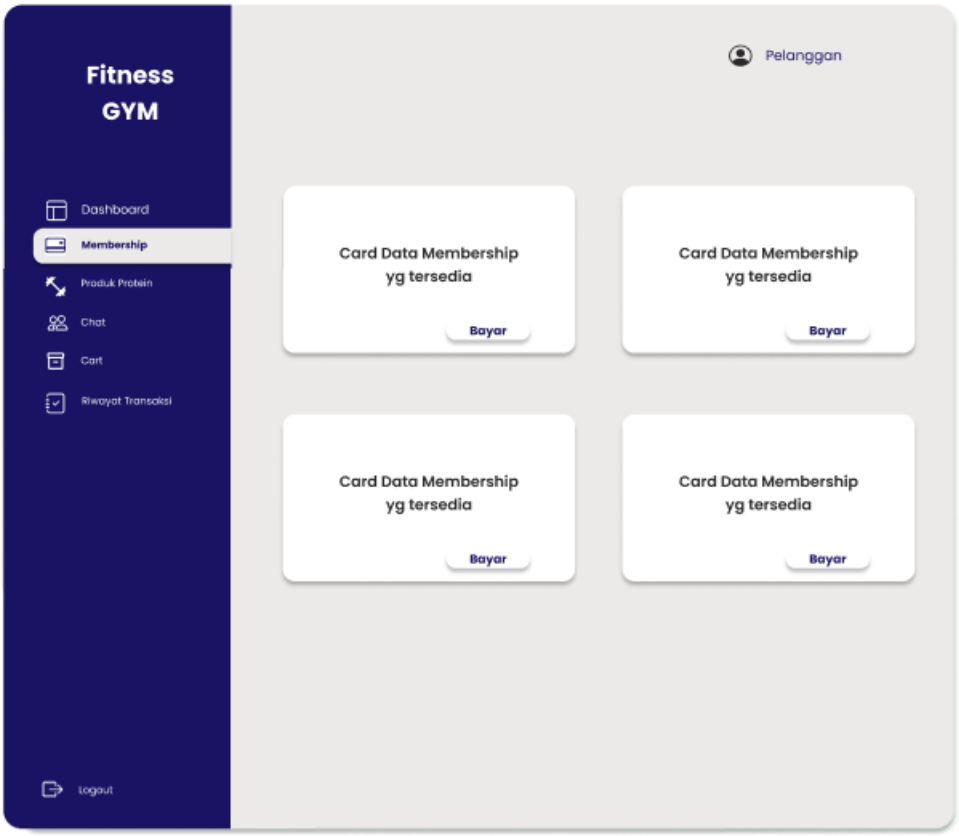


Deskripsi : Antarmuka Kelola Transaksi digunakan oleh administrator untuk melihat dan mengelola data transaksi yang terjadi dalam sistem. Administrator memiliki kewenangan untuk melakukan konfirmasi pembayaran transaksi hingga transaksi dinyatakan selesai. Setiap transaksi memiliki status yang menunjukkan tahapan proses pembayaran.

Input	:	Kata kunci pencarian data transaksi Aksi konfirmasi pembayaran
Output	:	Data transaksi ditampilkan dalam tabel data transaksi Perubahan status transaksi Pesan hasil konfirmasi pembayaran
Method/Algoritma	:	On Load 'Kelola Transaksi' 1. Sistem menampilkan halaman kelola transaksi. 2. Sistem mengambil seluruh data transaksi dari tabel Transaksi. 3. Sistem menampilkan data transaksi dalam bentuk tabel data transaksi beserta status transaksi. On Click 'Konfirmasi Pembayaran' 1. Sistem menampilkan detail transaksi yang dipilih. 2. Administrator melakukan konfirmasi pembayaran. 3. Sistem memperbarui status transaksi menjadi "Dikonfirmasi" pada tabel Transaksi. 4. Sistem menampilkan pesan "Pembayaran berhasil dikonfirmasi".

5. Sistem memperbarui tampilan tabel data transaksi.

9.Antarmuka Pelanggan *Membership*

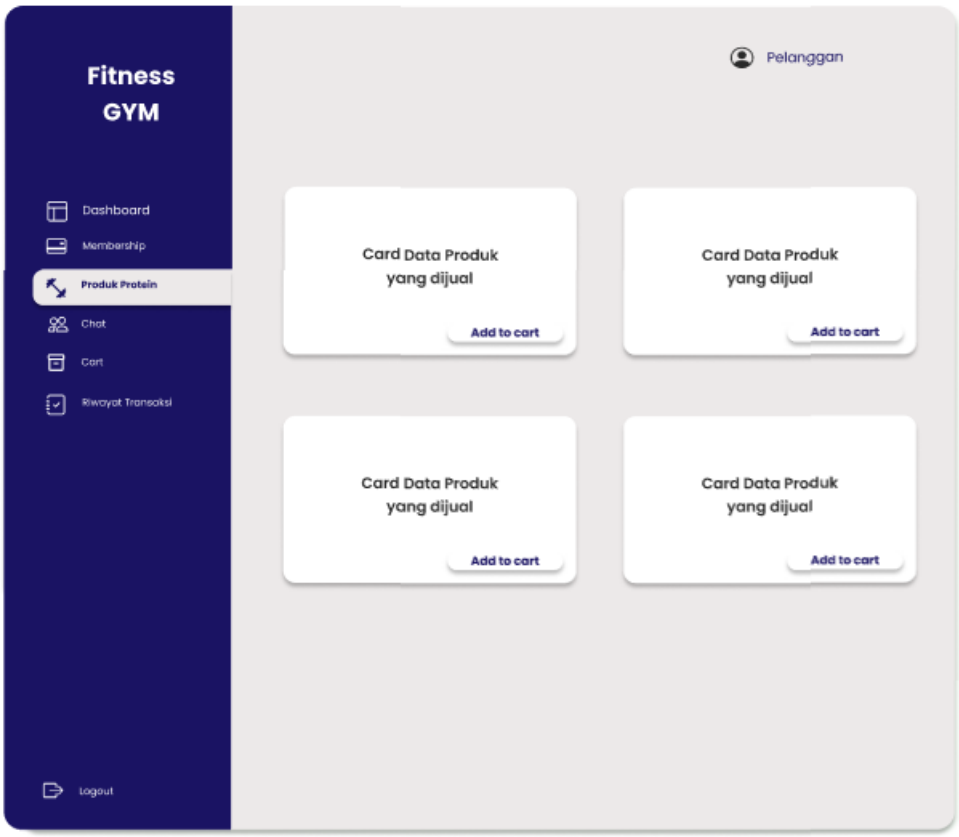


Deskripsi : Antarmuka *membership* digunakan oleh pelanggan untuk melihat daftar paket *membership* gym yang tersedia dalam sistem informasi. Pelanggan dapat menelusuri dan memilih paket *membership* yang diinginkan, kemudian menambahkannya ke keranjang pembelian. Antarmuka ini menampilkan informasi paket *membership* secara ringkas dalam bentuk kartu (card).

Input : Pilihan paket *membership*
Aksi Add to Cart

Output	:	Daftar paket *membership* ditampilkan dalam bentuk card Paket pelangganship berhasil ditambahkan ke keranjang Pesan notifikasi hasil aksi pelanggan
Method/Algoritma	:	On Load 'Page Pelangganship' *1.* Sistem menampilkan halaman *membership* pelanggan. *2.* Sistem menampilkan data *membership* dalam bentuk card data *membership*. On Click 'Bayar' *1.* Sistem membaca data *membership* yang dipilih pelanggan. *2.* Sistem melakukan validasi data *membership*. *3.* Sistem langsung berpindah ke halaman transaksi.

10. Antarmuka Pelanggan Produk



Deskripsi : Antarmuka Produk Protein digunakan oleh pelanggan untuk melihat daftar produk protein yang dijual oleh gym. Pelanggan dapat menelusuri produk yang tersedia dan menambahkan produk yang dipilih ke dalam keranjang pembelian. Antarmuka ini menampilkan informasi produk dalam bentuk kartu yang memuat detail singkat produk.

Input : Pilihan produk protein
Aksi Add to Cart

Output : Daftar produk protein ditampilkan dalam bentuk card

Produk protein berhasil ditambahkan ke keranjang
Pesan notifikasi hasil aksi pelanggan

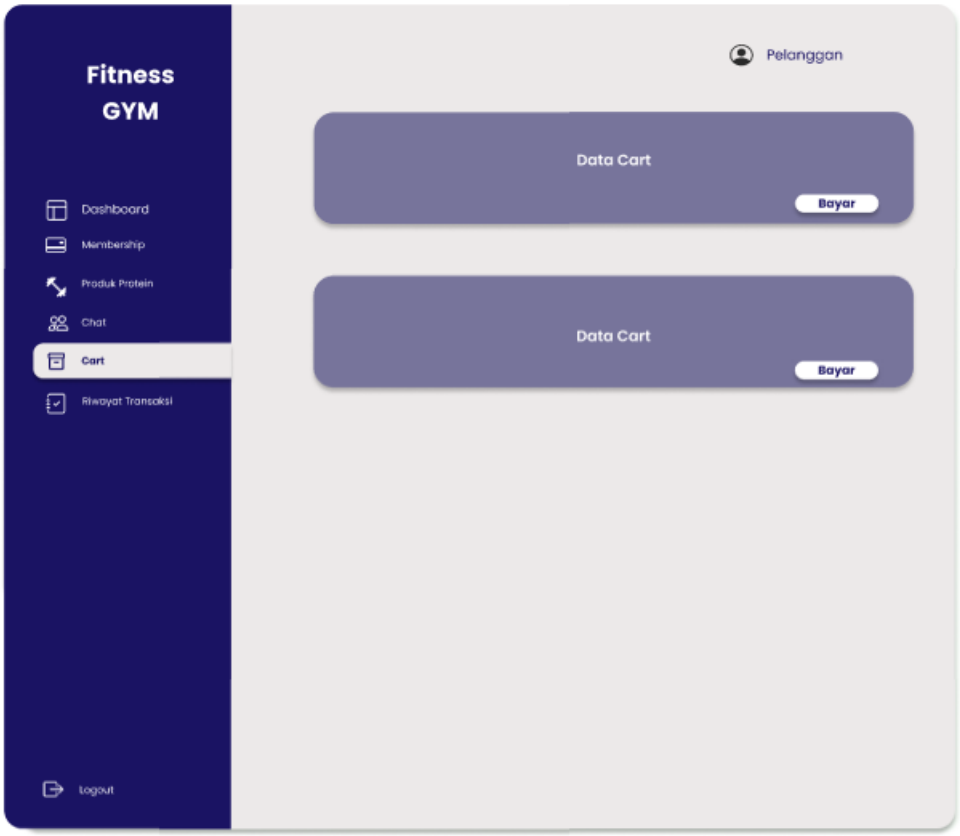
Method/Algoritma : On Load ‘Produk Protein’

1. Sistem menampilkan halaman produk protein pelanggan.
2. Sistem mengambil seluruh data produk protein yang berstatus aktif dari tabel Produk.
3. Sistem menampilkan data produk protein dalam bentuk card data produk.

On Click ‘Add to Cart’

1. Sistem membaca data produk protein yang dipilih pelanggan.
2. Sistem melakukan validasi ketersediaan produk dan stok.
3. Jika produk tidak tersedia atau stok habis, tampilkan pesan “Produk tidak tersedia”, kemudian kembali ke langkah 1.
4. Jika data valid, sistem menambahkan data produk ke tabel Cart.
5. Sistem mengurangi jumlah stok produk sementara.
6. Sistem menampilkan pesan “Produk berhasil ditambahkan ke keranjang”.

11. Antarmuka Pelanggan *Cart*



Deskripsi : Antarmuka Cart digunakan oleh pelanggan untuk melihat dan mengelola daftar item yang telah ditambahkan ke dalam keranjang belanja. Pelanggan dapat mengubah jumlah item, menghapus item tertentu, serta melanjutkan proses pembayaran. Antarmuka ini menampilkan data item cart secara ringkas sebelum dilakukan transaksi.

Input : Aksi tambah / kurangi jumlah item
Aksi hapus item
Aksi bayar

Output : Data cart ditampilkan dalam daftar cart
Perubahan jumlah item pada cart
Item terhapus dari cart

Pesan notifikasi hasil aksi pelanggan

Method/Algoritma : On Load ‘Page Cart’

1. Sistem menampilkan halaman cart pelanggan.
2. Sistem mengambil seluruh data cart milik pelanggan dari tabel Cart.
3. Sistem menampilkan data cart dalam bentuk daftar item cart beserta jumlah dan harga.

On Click ‘Tambah Jumlah’

1. Sistem membaca data item yang dipilih.
2. Sistem menambahkan jumlah item sebanyak satu.
3. Sistem melakukan validasi stok.
4. Jika stok tidak mencukupi, tampilkan pesan “Stok tidak mencukupi” dan batalkan perubahan.
5. Sistem memperbarui jumlah item pada tabel Cart.
6. Sistem memperbarui total harga cart.

On Click ‘Kurangi Jumlah’

1. Sistem membaca data item yang dipilih.
2. Sistem mengurangi jumlah item sebanyak satu.
3. Jika jumlah item menjadi nol, sistem menghapus item dari cart.
4. Sistem memperbarui data cart dan total harga.

On Click ‘Hapus Item’

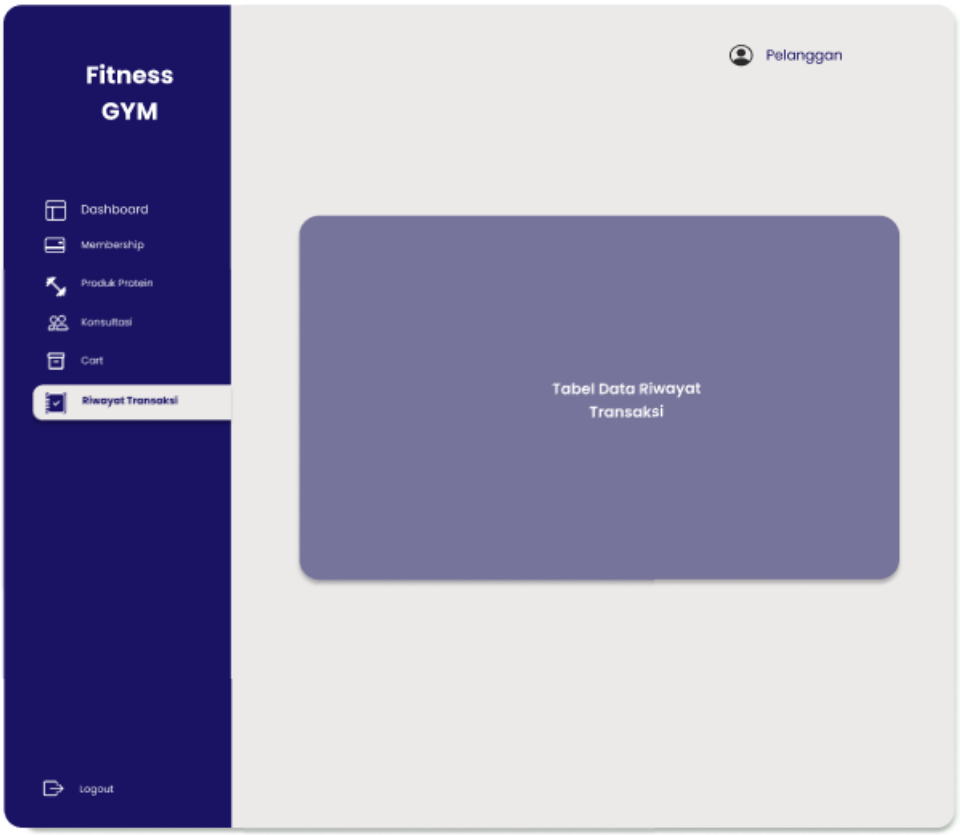
1. Sistem menampilkan konfirmasi penghapusan item cart.
2. Jika pelanggan menyetujui, sistem menghapus item dari tabel Cart.
3. Sistem menampilkan pesan “Item berhasil dihapus dari cart”.
4. Sistem memperbarui tampilan data cart.

On Click ‘Bayar’

1. Sistem membaca seluruh data cart pelanggan.
2. Sistem menghitung total pembayaran.

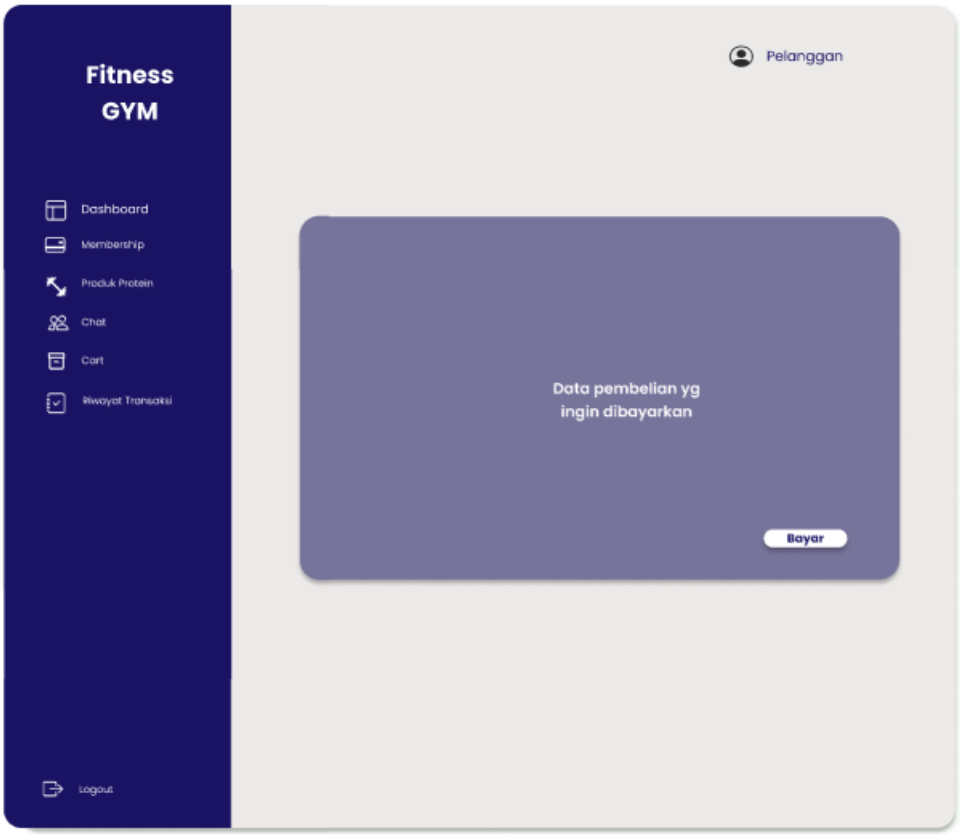
3. Sistem membuat data transaksi baru pada tabel Transaksi.
4. Sistem mengubah status transaksi menjadi "Menunggu Konfirmasi".
5. Sistem menampilkan pesan "Transaksi berhasil dibuat, menunggu konfirmasi pembayaran".
6. Sistem mengosongkan data cart pelanggan.

12. Antarmuka Pelanggan Riwayat Transaksi



Deskripsi	:	Antarmuka Riwayat Transaksi digunakan oleh pelanggan untuk melihat daftar transaksi yang pernah dilakukan dalam sistem. Antarmuka ini menampilkan informasi transaksi seperti tanggal transaksi, jenis transaksi, total pembayaran, dan status transaksi. Riwayat transaksi membantu pelanggan dalam memantau status pembayaran dan histori pembelian *membership* maupun produk.
Input	:	Kata kunci pencarian data transaksi
Output	:	Data riwayat transaksi ditampilkan dalam tabel riwayat transaksi Informasi detail status transaksi
Method/Algoritma	:	On Load 'Page Riwayat ' *1.* Sistem menampilkan halaman riwayat transaksi pelanggan. *2.* Sistem mengambil seluruh data transaksi milik pelanggan yang sedang login dari tabel Transaksi. *3.* Sistem menampilkan data transaksi dalam bentuk tabel riwayat transaksi.

13. Antarmuka Pelanggan Transaksi



Deskripsi	:	Antarmuka Transaksi digunakan oleh pelanggan untuk melakukan proses pembayaran terhadap pembelian maupun produk protein yang telah dipilih. Antarmuka ini menampilkan ringkasan data pembelian yang akan dibayarkan serta status transaksi sebelum dan sesudah pembayaran dilakukan.
Input	:	Data pembelian yang akan dibayarkan Metode pembayaran Aksi konfirmasi pembayaran
Output	:	Data transaksi tersimpan di sistem Status transaksi “Menunggu Konfirmasi”

Pesan pembayaran berhasil dikirim

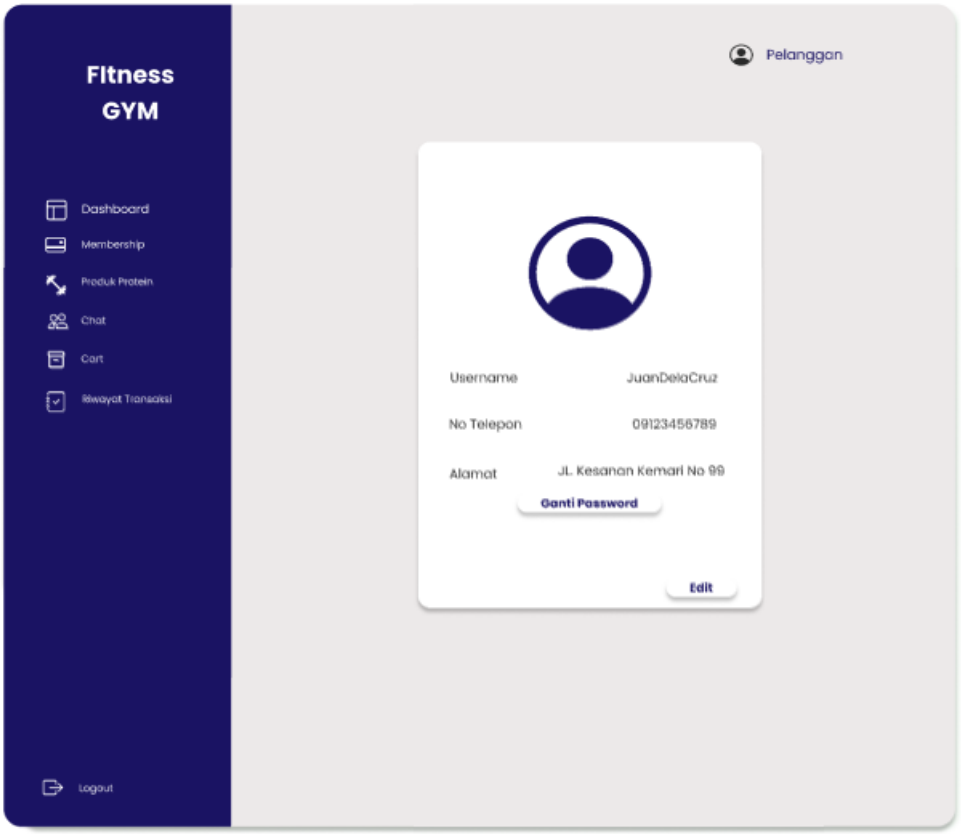
Method/Algoritma : On Load ‘Page Transaksi’

1. Sistem menampilkan halaman transaksi pelanggan.
2. Sistem mengambil data pembelian dari tabel Transaksi dengan status belum dibayar.
3. Sistem menampilkan ringkasan data pembelian yang akan dibayarkan.

On Click ‘Bayar’

1. Sistem membaca data transaksi yang dipilih pelanggan.
2. Sistem melakukan validasi data transaksi.
3. Jika data tidak valid atau kosong, tampilkan pesan “Data transaksi tidak valid”, kembali ke halaman sebelumnya.
4. Jika data valid, sistem menyimpan data pembayaran.
5. Sistem mengubah status transaksi menjadi “Menunggu Konfirmasi”.
6. Sistem menampilkan pesan “Pembayaran berhasil dikirim, menunggu konfirmasi”.

14. Antarmuka Pelanggan Edit Profil



Deskripsi	:	Antarmuka Edit Profil digunakan oleh pelanggan untuk melihat dan memperbarui data profil pribadi yang tersimpan dalam sistem. Pelanggan dapat mengubah informasi seperti nomor telepon dan alamat, serta melakukan perubahan kata sandi akun. Antarmuka ini bertujuan untuk memastikan data pelanggan selalu akurat dan terkini.
Input	:	Nomor telepon Alamat Password lama (khusus perubahan password) Password baru (khusus perubahan password)
Output	:	Data profil pelanggan berhasil diperbarui Pesan keberhasilan atau kegagalan perubahan data
Method/Algoritma	:	On Load ‘Page Edit Profil’ *1.* Sistem menampilkan halaman edit profil pelanggan.

2. Sistem mengambil data profil pelanggan yang sedang login dari tabel Pelanggan.
3. Sistem menampilkan data profil pelanggan dalam bentuk informasi profil.

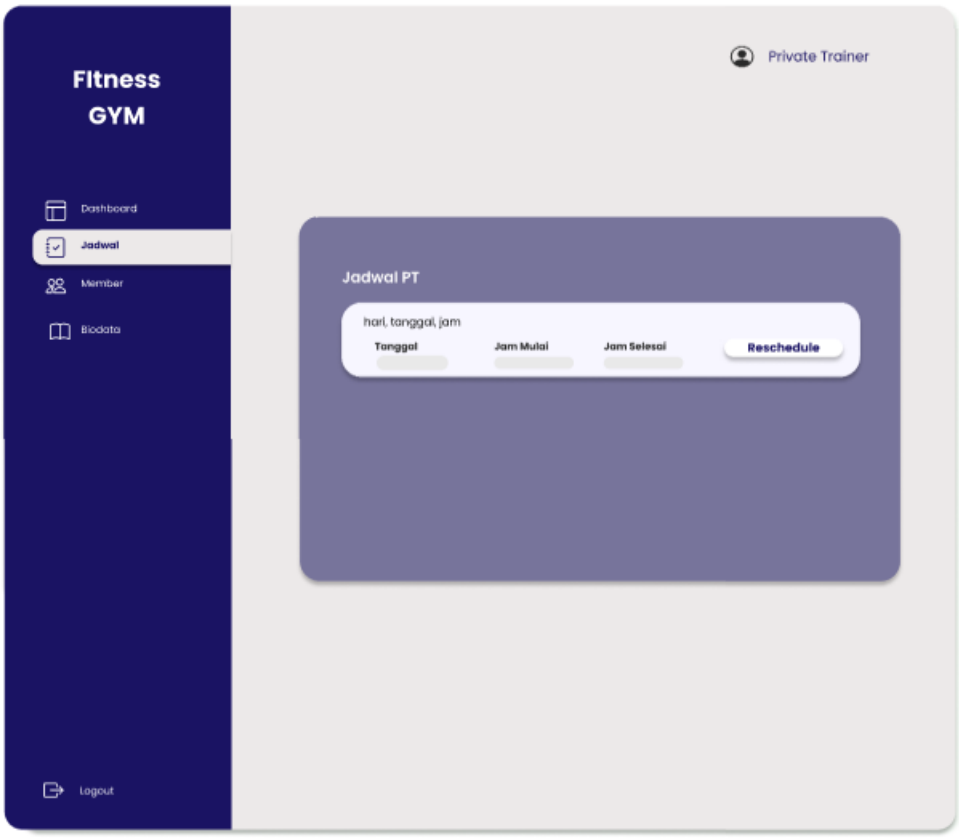
On Click 'Edit'

1. Sistem mengaktifkan form edit profil.
2. Pelanggan mengubah data nomor telepon dan/atau alamat.
3. Sistem membaca data input yang diubah.
4. Sistem melakukan validasi data input.
5. Jika terdapat data tidak valid, tampilkan pesan "Ada data yang belum diisi atau tidak valid", kembali ke langkah 2.
6. Jika data valid, sistem menyimpan perubahan data ke tabel Pelanggan.
7. Sistem menampilkan pesan "Data profil berhasil diperbarui".

On Click 'Ganti Password'

1. Sistem menampilkan form ganti password.
2. Pelanggan memasukkan password lama dan password baru.
3. Sistem memvalidasi kesesuaian password lama.
4. Jika password lama tidak sesuai, tampilkan pesan "Password lama tidak sesuai", kembali ke langkah 2.
5. Jika valid, sistem memperbarui password pelanggan.
6. Sistem menampilkan pesan "Password berhasil diperbarui".

15. Antarmuka *Private trainer* Jadwal



Deskripsi : Antarmuka Jadwal Pelatihan digunakan oleh *Private trainer* untuk melihat jadwal pelatihan yang diajukan user. Antarmuka ini menampilkan daftar jadwal pelatihan yang harus dilaksanakan oleh *Private trainer* beserta informasi pelanggan, tanggal, dan waktu pelatihan. Jadwal pelatihan dapat reschedule oleh *private trainer* dan pengguna pelanggan dengan syarat konfirmasi dari *private trainer*

Input : Masukan tanggal, jam mulai dan jam selesai, kemudian *private trainer* dapat reschedule jadwal yang dibuat pelanggan

Output : Data jadwal pelatihan ditampilkan dalam tabel jadwal pelatihan

Informasi detail jadwal pelatihan

Method/Algoritma : On Load 'Page Jadwal Pelatihan'

1. Sistem menampilkan halaman jadwal pelatihan *Private trainer*.

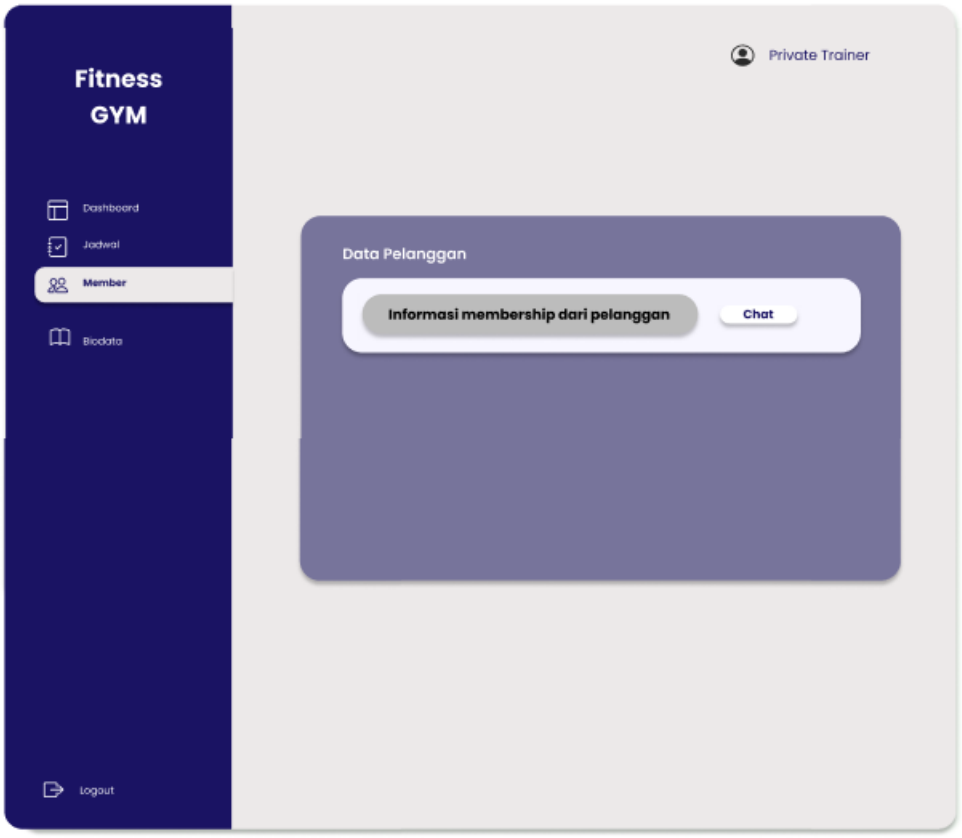
2. Sistem mengambil data jadwal pelatihan dari tabel Jadwal berdasarkan *Private trainer* yang sedang login.

3. Sistem menampilkan data jadwal pelatihan dalam bentuk tabel jadwal pelatihan.

On Click '*Reschedule*'

1. Sistem menampilkan data *user* yang diajukan.
2. *Private trainer* menginputkan tanggal, jam mulai dan jam selesai.
3. Klik tombol *Reschedule*
4. Sistem memperbarui status jadwal pelatihan.

16. Antarmuka *Private trainer* Pelanggan



Deskripsi : Antarmuka Pelanggan digunakan oleh *Private trainer* untuk melihat daftar pelanggan yang memilih atau terdaftar pada *Private trainer* tersebut. *Private trainer* dapat mengirim pesan kepada user terkait dengan pelangganship yang diambil. Antarmuka ini membantu *Private trainer* dengan pelanggan dalam mengelola jadwal pelatihan secara terstruktur maupun menanyakan sekitar pelatihan.

Input : Pesan yang ingin dikirim

Output : Data pelanggan tertampil pada halaman dengan ada tombol chat untuk menamilkan chat

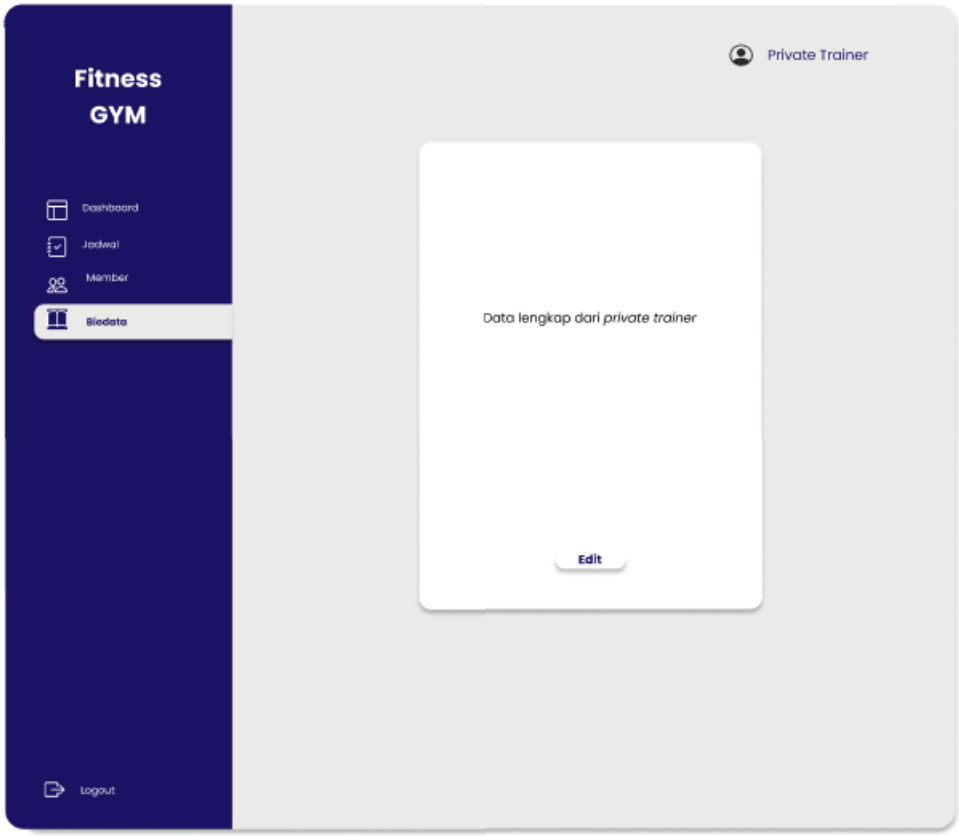
Method/Algoritma : On Load ‘ Pelanggan *Private trainer*’

1. Sistem menampilkan halaman pelanggan *Private trainer*.
2. Sistem mengambil data pelanggan yang terdaftar pada *Private trainer* yang sedang login dari tabel Pelanggan *Private trainer*.
3. Sistem menampilkan data pelanggan dalam bentuk tabel pelanggan.

On Click '*Chat*'

1. *Private trainer* memulai mengirim pesan dengan pelanggan.
2. Pesan terikirim kepada pelanggan.
3. Pelanggan membalas pesan dan tertampil pada halaman pesan *private trainer*.

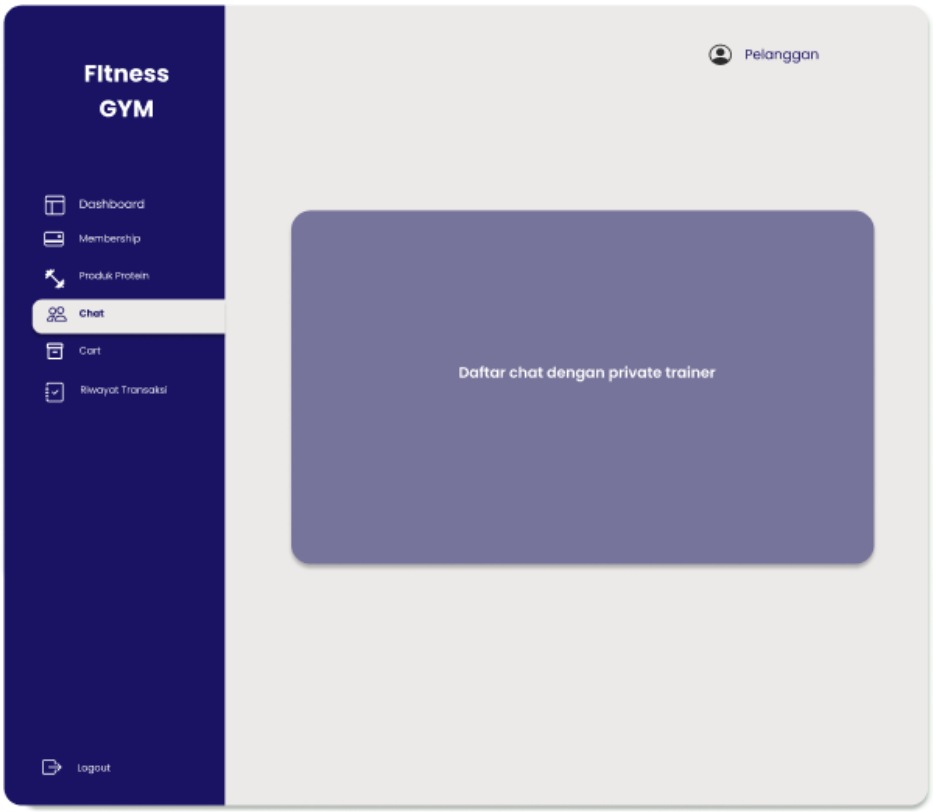
17. Antarmuka *Private trainer* Pelanggan



Deskripsi	:	Antarmuka Pelanggan digunakan oleh *Private trainer* untuk melihat data profil lengkap *private trainer*. *Private trainer* dapat mengisi dan menubah data informasi lengkap disini. Antarmuka ini membantu pelanggan dalam memilih *private trainer* yang tersedia dengan melihat informasi lengkap apabila memilih paket bundle.
Input	:	Informasi data lengkap *private trainer*
Output	:	Data informasi *private trainer* tertampil pada halaman ini
Method/Algoritma	:	On Load ‘ biodata *Private trainer*’

1. Sistem menampilkan halaman pelanggan *Private trainer*.
2. Sistem mengambil data user *private trainer* yang telah dibuat oleh admin.
3. *Private trainer* dapat mengubah data yang telah ada.

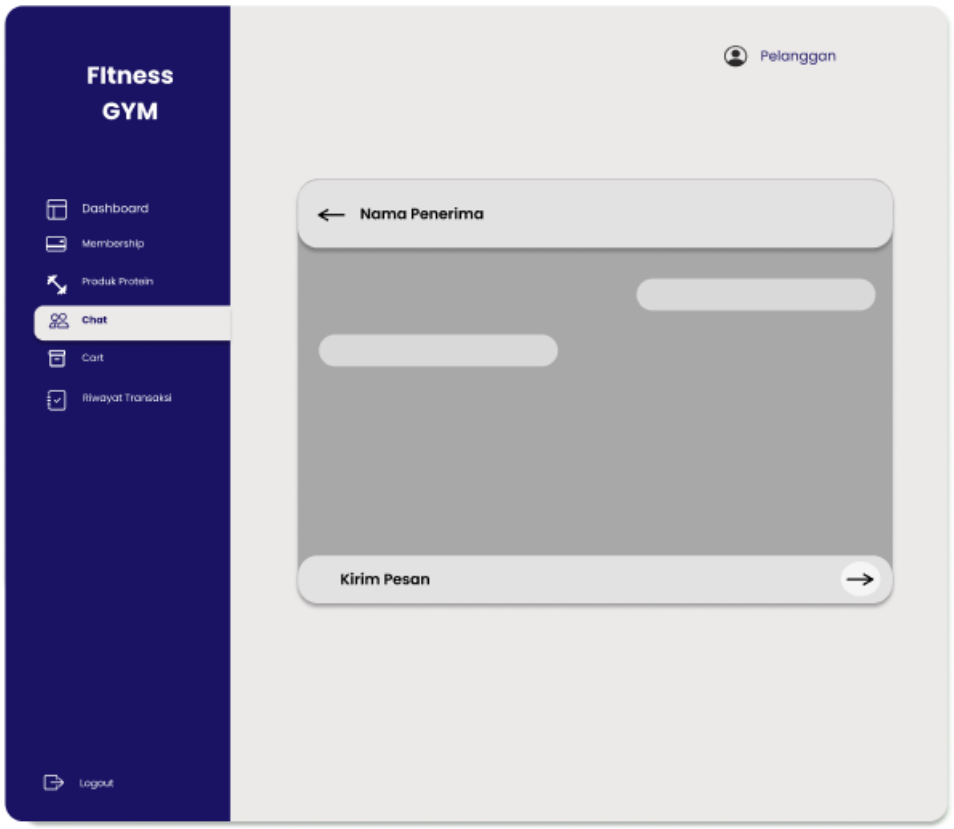
18. Antarmuka Pelanggan *Chat*



Deskripsi	:	Antarmuka Pelanggan digunakan oleh pelanggan untuk melakukan percakapan / chat dengan *private trainer*. Halaman ini menampilkan daftar percakapan dengan *private trainer*. Tekan pada salah satu daftar maka detail percakapan akan terlihat dan tertampil pada halaman ini.
Input	:	Daftar Percakapan dengan *private trainer*.
Output	:	Setelah menulis pesan yang ingin dikirim makan pesan tersebut akan tersampaikan pada *private trainer* yang dituju
Method/Algoritma	:	On Load ‘ Chat’ *1.* Sistem menampilkan daftar percakapan dengan *private trainer*.

2. Sistem menampilkan pesan lengkap dengan *private trainer* yang ingin dituju.
3. *Private trainer* akan menapatkan pesan dari pelanggan setelah pesan dikirim.

19. Antarmuka Chat



Deskripsi : Antarmuka Pelanggan digunakan oleh pelanggan untuk melakukan percakapan / chat dengan *private trainer*. Halaman ini menampilkan tampilan chat antara pengirim dan penerima.

Input	:	Isi percakapan dengan *private trainer*.
Output	:	Setelah menulis pesan yang ingin dikirim makan pesan tersebut akan tersampaikan pada *private trainer* yang dituju
Method/Algoritma	:	On Load ‘ Chat’ 1. Sistem menampilkan pesan lengkap dengan *private trainer* yang ingin dituju. 2. *Private trainer* akan mendapatkan pesan dari pelanggan setelah pesan dikirim.

PERENCANAAN DAN DESKRIPSI HASIL UJI PERANGKAT LUNAK

PEMBANGUNAN SISTEM INFORMASI BERBASIS *WEBSITE*

Disusun oleh:

Axel Christian Nugroho

210711114

Program Studi In*form*atika

Fakultas Teknologi Industri

Universitas Atma Jaya Yogyakarta

	Program Studi Teknik	Nomor Dokumen	Halaman

UNIVERSITAS ATMA JAYA YOGYAKARTA

DAFTAR REVISI

Revisi	Deskripsi
A	
B	
C	
D	
E	

F	

INDEX TGL	-	A	B	C	D	E	F
Ditulis oleh							
Diperiksa oleh							
Disetujui oleh							

Daftar Isi

DAFTAR REVISI 216

1. Pendahuluan 1

1.1 Tujuan 1

1.2 Deskripsi Umum Sistem 1

1.3 Definisi, Akronim, dan Singkatan 3

1.4 Referensi 5

2. Lingkungan Pengujian Perangkat Lunak 6

2.1 Perangkat Lunak Pengujian 6

2.2 Perangkat Keras Pengujian 7

2.3 Material Pengujian 8

2.4 Sumber Daya Manusia 9

2.5 Prosedur Umum Pengujian 10

2.5.1. Pengenalan dan Latihan 10

2.5.2. Persiapan Awal 10

2.5.3. Pelaksanaan 13

2.5.4. Pelaporan Hasil 14

3. Identifikasi dan Rencana Pengujian 14

3.1. Identifikasi Pengujian 15

3.2. Rencana Pengujian 19

4. Deskripsi Pengujian 23

4.1 Identifikasi Kelas Pengujian *Use Case Login* 23

4.1.1 Identifikasi Butir Pengujian Login Sistem (AU-001) 23

4.1.2 Identifikasi Butir Pengujian Logout Sistem (AU-001-02) 23

4.2 Identifikasi Kelas Pengujian *Use Case* Registrasi 23
4.2.1 Identifikasi Butir Pengujian Registrasi Pengguna (AU-002) 23
4.3. Identifikasi Kelas Pengujian Kelola *Membership*.....23
4.3.1 Identifikasi Butir Pengujian Tambah *Membership* (AU-101-01) 23
4.3.2 Identifikasi Butir Pengujian Edit *Membership* (AU-101-02) 24
4.3.3 Identifikasi Butir Pengujian Hapus *Membership* (AU-101-03) 24
4.3.4 Identifikasi Butir Pengujian Lihat Data *Membership* (AU-101-04) 24
4.4 Identifikasi Kelas Pengujian Kelola Produk 24
4.4.1 Identifikasi Butir Pengujian Tambah Produk (AU-201-01) 24
4.4.2 Identifikasi Butir Pengujian Edit Produk (AU-201-02) 24
4.4.4 Identifikasi Butir Pengujian Lihat Data Produk (AU-201-04) 24
4.5 Identifikasi Kelas Pengujian Kelola Data Pengguna 25
4.5.1 Identifikasi Butir Pengujian Hapus Data Pengguna (AU-501-03) 25
4.6 Identifikasi Kelas Pengujian Kelola *Private Trainer*.....25
4.6.1 Identifikasi Butir Pengujian Tambah *Private Trainer* (AU-701-01) 25
4.6.2 Identifikasi Butir Pengujian Hapus *Private Trainer* (AU-701-03) 25
4.6.3 Identifikasi Butir Pengujian Lihat Data *Private Trainer* (AU-701-04) 25
4.7 Identifikasi Kelas Pengujian Pembayaran *Membership*.....25
4.7.1 Identifikasi Butir Pengujian Proses Pembayaran *Membership* (PM-001) 25
4.7.2 Identifikasi Butir Pengujian Validasi Pembayaran *Membership* (PM-

002) 26

4.7.3 Identifikasi Butir Pengujian Kirim Bukti Pembayaran *Membership* (PM-003) 26

4.7.4 Identifikasi Butir Pengujian Riwayat Pembayaran *Membership* (PM-004) 26

4.8 Identifikasi Kelas Pengujian Pembayaran Produk 26

4.8.1 Identifikasi Butir Pengujian Proses Pembayaran Produk (PP-001) 26

4.8.2 Identifikasi Butir Pengujian Kirim Bukti Pembayaran Produk (PP-002) 26

4.8.3 Identifikasi Butir Pengujian Menunggu Konfirmasi Pembayaran Produk (PP-003) 27

4.8.4 Identifikasi Butir Pengujian Riwayat Pembayaran Produk (PP-004) 27

4.9 Identifikasi Kelas Pengujian Laporan 27

4.9.1 Identifikasi Butir Pengujian Menampilkan Laporan (LP-001) 27

4.9.2 Identifikasi Butir Pengujian Filter Laporan (LP-002) **Error! Bookmark not defined.**

4.9.3 Identifikasi Butir Pengujian Cetak Laporan (LP-003) **Error! Bookmark not defined.**

4.10 Identifikasi Kelas Pengujian *Use Case Edit* Profil**Error! Bookmark not defined.**

4.10.1 Identifikasi Butir Pengujian Menampilkan Data Profil (EP-001) **Error! Bookmark not defined.**

4.10.2 Identifikasi Butir Pengujian Edit Profil (EP-002) **Error! Bookmark not defined.**

4.10.3 Identifikasi Butir Pengujian Validasi Data Edit Profil (EP-003)**Error!**

Bookmark not defined.

4.10.4 Identifikasi Butir Pengujian Ubah Password (EP-004) **Error! Bookmark not defined.**

4.10.5 Identifikasi Butir Pengujian Simpan Perubahan Profil (EP-005) **Error! Bookmark not defined.**

4.11 Identifikasi Kelas Pengujian *Use Case* Jadwal Pelatihan ..... **Error! Bookmark not defined.**

4.11.1 Identifikasi Butir Pengujian Tambah Jadwal Pelatihan (AU-801-01) **Error! Bookmark not defined.**

4.11.2 Identifikasi Butir Pengujian Tampil Data Jadwal Pelatihan (AU-801-02) **Error! Bookmark not defined.**

4.11.3 Identifikasi Butir Pengujian Reschedule Jadwal Pelatihan (AU-801-03) **Error! Bookmark not defined.**

4.11.4 Identifikasi Butir Pengujian Validasi Bentrok Jadwal Pelatihan (AU-801-04) **Error! Bookmark not defined.**

4.11.5 Identifikasi Butir Pengujian Konfirmasi Jadwal Pelatihan (AU-801-05) **Error! Bookmark not defined.**

4.12 Identifikasi Kelas Pengujian Informasi *Private Trainer*..... **Error! Bookmark not defined.**

4.12.1 Identifikasi Butir Pengujian Tampil Data *Private Trainer* (IT-001) **Error! Bookmark not defined.**

4.12.2 Pengujian Tambah Data *Private Trainer* (IT-002) **Error! Bookmark not defined.**

4.12.3 Pengujian Ubah Data *Private Trainer* (IT-003) 25

5. Hasil Pengujian .....**Error! Bookmark not defined.**

1. Pendahuluan

1.1 Tujuan

Dokumen Perencanaan, Deskripsi, dan Hasil Uji Perangkat Lunak (PDHUPL) ini disusun untuk memberikan panduan dalam proses pengujian sistem yang dikembangkan, yaitu sistem informasi yang dibangun sebagai bagian dari tugas akhir. Tujuan utama dari penyusunan dokumen PDHUPL ini adalah untuk:

a. Menjelaskan rencana dan strategi pengujian perangkat lunak secara sistematis.
b. Menjadi acuan teknis bagi tim pengembang, khususnya penguji, dalam melakukan proses pengujian aplikasi.
c. Memastikan bahwa sistem yang dikembangkan telah berjalan sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan.
d. Mendokumentasikan hasil pengujian sebagai bahan evaluasi dan perbaikan sistem.

Dokumen ini ditujukan untuk digunakan oleh pihak-pihak yang terlibat dalam pengembangan perangkat lunak, terutama:

a. *Developer*, sebagai referensi dalam memahami alur pengujian.
b. *Tester*, sebagai pedoman dalam melakukan pengujian sistem.
c. Pembimbing atau penguji akademik, sebagai bahan evaluasi terhadap kualitas perangkat lunak yang dikembangkan.

1.2 Deskripsi Umum Sistem

Sistem yang dikembangkan merupakan perangkat lunak berbasis *web* yang digunakan untuk mengelola layanan operasional pada sistem informasi *gym*. Sistem ini dirancang untuk membantu proses manajemen data, transaksi, serta pelayanan kepada pelanggan. Secara umum, sistem ini memiliki beberapa modul utama sebagai berikut:

1. Manajemen Data pelanggan

   Modul ini digunakan untuk mengelola data pengguna sistem. Fitur yang disediakan meliputi penghapusan data pelanggan.

2. Manajemen Data *Private Trainer*

   Modul ini digunakan untuk mengelola data *private trainer*. Fitur yang disediakan meliputi penambahan data *user private trainer* dan penghapusan data pelanggan.

3. Manajemen Produk

   Modul ini memungkinkan admin untuk mengelola data produk susu protein yang tersedia. Pengelolaan meliputi penambahan produk, pengubahan informasi produk, serta penghapusan produk.

4. Manajemen *Membership*

   Modul ini digunakan untuk mengelola paket *membership* yang tersedia bagi pelanggan. Admin dapat menambahkan, mengubah, dan menghapus jenis *membership* beserta durasi dan harga yang ditawarkan.

5. Manajemen Jadwal

   Modul ini digunakan untuk mengelola jadwal pelatihan pelanggan yang memilih paket *bundle* dengan *private trainer*. pelanggan dapat membuat dan mengatur ulang jawal dengan persetujuan *private trainer. Private trainer* dapat menyetujui, menolak, dan mengatur ulang jadwal tanpa harus persetujuan pelanggan.

6. Transaksi Produk

   Modul ini digunakan untuk menangani proses pembelian produk oleh pelanggan. Sistem akan mencatat detail transaksi, menghitung total pembayaran, serta mengelola status transaksi.

7. Transaksi *Membership*

   Modul ini digunakan untuk menangani pendaftaran atau perpanjangan

membership pelanggan. Sistem akan mencatat data transaksi dan status *membership* pelanggan aktif setelah melakukan transaksi.

8. Manajemen Keranjang (*Cart*)
   Modul ini memungkinkan pelanggan untuk menambahkan produk ke dalam keranjang sebelum melakukan transaksi. Data pada keranjang akan disimpan sementara sebelum diproses menjadi transaksi.
9. Verifikasi dan Validasi Transaksi
   Modul ini digunakan oleh admin untuk melakukan verifikasi terhadap transaksi yang dilakukan, seperti menyetujui atau menolak transaksi, serta memastikan data transaksi valid.
10. Laporan dan Monitoring
    Modul ini menyediakan informasi laporan terkait transaksi produk dan *membership*, yang dapat digunakan untuk analisis dan pengambilan keputusan.

1.3 Definisi, Akronim, dan Singkatan

Subbagian ini menjelaskan istilah-istilah penting, definisi, serta singkatan yang digunakan dalam dokumen PDHUPL agar tidak terjadi kesalahpahaman dalam interpretasi.

Definisi	
Sistem	Aplikasi berbasis *web* yang digunakan untuk mengelola operasional layanan *gym*, termasuk data pelanggan, produk, dan *membership*.
Pengguna	Pihak yang menggunakan sistem, baik internal (admin, *private trainer*) maupun eksternal (pelanggan).
Produk	Barang yang dijual dalam sistem, seperti suplemen.
Membership	Layanan berlangganan yang memberikan akses kepada pelanggan untuk menggunakan fasilitas *gym* dalam periode tertentu.
Transaksi Produk	Proses pembelian produk oleh pelanggan yang dicatat dalam sistem.
Transaksi *Membership*	Proses pendaftaran atau perpanjangan *membership* pelanggan.
Keranjang (*Cart*)	Fitur yang digunakan untuk menampung sementara produk yang akan dibeli sebelum diproses menjadi transaksi.
Verifikasi Transaksi	Proses pengecekan dan persetujuan transaksi oleh admin untuk memastikan keabsahan data.
Private Trainer	Pelatih pribadi yang memberikan bimbingan dan program latihan secara khusus kepada pelanggan dalam sistem *gym*. *Private trainer* bertanggung jawab dalam menyusun jadwal latihan, memantau perkembangan, serta membantu pelanggan mencapai tujuan kebugaran secara lebih terarah.

Akronim dan Singkatan	
PDHUPL	Perencanaan, Deskripsi, dan Hasil Uji Perangkat Lunak
DPPL	Deskripsi Perancangan Perangkat Lunak
UI (User Interface)	Tampilan antarmuka yang digunakan oleh pengguna untuk berinteraksi dengan sistem
DB (*Database*)	Kumpulan data yang tersimpan secara terstruktur dalam sistem

1.4 Referensi

1. Laporan Tugas Akhir Pembangunan system informasi berbasis *website*
2. SKPL Pembangunan system informasi berbasis *website*
3. DPPL Pembangunan system informasi berbasis *website*

2. Lingkungan Pengujian Perangkat Lunak

2.1 Perangkat Lunak Pengujian

Subbagian ini menjelaskan spesifikasi perangkat lunak yang digunakan dalam proses pengujian sistem, serta data yang digunakan untuk mendukung pengujian.

Perangkat lunak yang digunakan dalam pengujian sistem ini adalah sebagai berikut:

1. Sistem Operasi

   Menggunakan Microsoft Windows 11 64-bit sebagai lingkungan utama dalam menjalankan aplikasi dan *tools* pengujian.

2. *Web Server*

   Menggunakan Apache sebagai *server* lokal untuk menjalankan aplikasi berbasis *web*.

3. Bahasa Pemrograman dan *Framework*

   Sistem dikembangkan menggunakan PHP dengan *framework* Laravel untuk mempermudah pengembangan dan pengujian aplikasi.

4. *Database Management Sistem* (DBMS)

   Menggunakan MySQL sebagai sistem pengelolaan basis data.

5. *Web Browser*

   Pengujian dilakukan menggunakan *browser* seperti Google Chrome untuk memastikan kompatibilitas tampilan dan fungsi sistem.

Data Pengujian

Data yang digunakan dalam proses pengujian meliputi:

1. Data Pengguna: Data admin, *private trainer*, dan pelanggan yang digunakan untuk menguji hak akses sistem.
2. Data Produk: Data produk seperti nama produk, harga, dan stok untuk menguji transaksi produk.

3. Data *Membership*: Data paket *membership* untuk menguji proses pendaftaran dan perpanjangan.
4. Data Transaksi: Data transaksi produk dan *membership* untuk menguji alur sistem secara menyeluruh.
5. Data Keranjang: Data sementara yang digunakan dalam proses simulasi pembelian produk.

2.2 Perangkat Keras Pengujian

Subbagian ini menjelaskan spesifikasi perangkat keras yang digunakan dalam proses pengujian sistem untuk memastikan aplikasi dapat berjalan dengan baik pada lingkungan yang digunakan.

Perangkat keras yang digunakan untuk pengujian adalah sebagai berikut:

1. **Komputer / *Laptop* (*Server* & Development)**

   Digunakan untuk menjalankan *server* lokal dan basis data.

   Spesifikasi minimal:

   a. Prosesor: Intel Core i7 atau setara
   b. RAM: Minimal 8 GB
   c. Penyimpanan: Minimal 512 GB SSD

2. **Komputer / *Laptop* (Client / Pengguna)**

   Digunakan untuk mengakses sistem melalui *web browser*.

   Spesifikasi minimal:

   a. Prosesor: Intel Core i3 atau setara
   b. RAM: Minimal 4 GB
   c. Penyimpanan: Minimal 128 GB

3. **Perangkat Jaringan**

   Digunakan untuk menghubungkan *server* dan *client* dalam proses pengujian.

   a. Router atau hotspot

b. Koneksi internet (opsional, jika menggunakan *server* online)

2.3 Material Pengujian

Subbagian ini menjelaskan material yang digunakan dalam proses pengujian sistem, khususnya yang berkaitan dengan hasil keluaran (*output*) dari aplikasi.

Material yang digunakan dalam pengujian meliputi:

1. **Contoh Hasil Keluaran (*Print-out*)**

   Digunakan untuk menguji apakah sistem dapat menghasilkan output yang sesuai dengan kebutuhan. Contoh hasil keluaran yang digunakan antara lain:

   a. Struk pembayaran (invois) transaksi produk

   b. Bukti transaksi *membership*

   c. Laporan transaksi (produk dan *membership*)

2. **Media Cetak**

   Digunakan untuk menguji hasil cetakan dari sistem:

   a. Kertas ukuran A4

   b. Kertas invois

   c. Kertas dengan berbagai ketebalan

3. **Dokumentasi Sistem**

   a. Manual penggunaan sistem (user manual)

   b. Dokumen perancangan sistem (DPPL)

   c. Dokumen kebutuhan sistem

4. **Data Uji (Test Data)**

   a. Data pengguna (admin, *private trainer*, pelanggan)

   b. Data produk dan stok

   c. Data *membership*

   d. Data transaksi

2.4 Sumber Daya Manusia

Subbagian ini menjelaskan sumber daya manusia yang terlibat dalam proses pengujian sistem beserta peran dan spesifikasinya.

Sumber daya manusia yang terlibat dalam pengujian sistem adalah sebagai berikut:

1. ***Tester***

   Bertugas melakukan pengujian sistem sesuai dengan skenario uji yang telah dirancang, mencatat hasil pengujian, serta melaporkan bug atau kesalahan sistem.

   Jumlah : 2 – 3 orang

   Spesifikasi:

   a. Pendidikan minimal D3/S1 Teknik Informatika atau bidang terkait
   b. Memahami dasar pengujian perangkat lunak (software testing)
   c. Memiliki pengalaman dalam pengujian sistem

2. ***Developer***

   Bertugas memperbaiki bug atau kesalahan yang ditemukan selama proses pengujian, serta memastikan sistem berjalan sesuai dengan kebutuhan.

   Jumlah : 1 Orang

   Spesifikasi:

   a. Menguasai bahasa pemrograman yang digunakan (misalnya PHP/Laravel)
   b. Memahami struktur sistem yang dikembangkan
   c. Mampu melakukan debugging dan troubleshooting

3. **Pengguna**

   Bertugas melakukan uji coba sistem dari sisi pengguna untuk memasitikan sistem mudah digunakan.

   Jumlah : Min 30 orang

Spesifikasi:

a. Mewakili pengguna sistem

b. Mampu mengikuti skenario penggunaan sistem

2.5 Prosedur Umum Pengujian

2.5.1. Pengenalan dan Latihan

Subbagian ini menjelaskan mekanisme pengenalan dan pelatihan penggunaan sistem sebelum dilakukan proses pengujian.

Pengenalan dan pelatihan sistem dilakukan sebelum tahap pengujian dimulai, dengan tujuan agar pengguna memahami alur dan fungsi sistem yang akan diuji. Kegiatan ini dilaksanakan sekitar 1 minggu sebelum pengujian utama dilakukan.

Pengenalan sistem diberikan kepada pihak yang terlibat, yaitu admin, *private trainer*, dan pelanggan. Proses pengenalan dilakukan dengan cara sebagai berikut:

a. Memberikan penjelasan mengenai fitur-fitur utama sistem, seperti manajemen produk, *membership*, dan transaksi.
b. Demonstrasi penggunaan sistem secara langsung oleh pengembang.
c. Praktik langsung oleh pengguna dengan menggunakan sistem.

Selain itu, disediakan juga dokumentasi berupa user manual yang dapat digunakan sebagai panduan dalam mengoperasikan sistem. Dokumentasi ini dapat diakses dalam bentuk file digital. Melalui kegiatan pengenalan dan pelatihan ini, diharapkan seluruh pengguna yang terlibat dalam pengujian dapat memahami cara kerja sistem sehingga proses pengujian dapat berjalan dengan efektif dan sesuai dengan skenario yang telah direncanakan.

2.5.2. Persiapan Awal

2.5.2.1. Persiapan Prosedural

Subbagian ini menjelaskan proses persiapan yang berkaitan dengan prosedur pelaksanaan pengujian perangkat lunak. Persiapan prosedural pengujian diawali

dengan melakukan koordinasi antara pihak pengembang dan pihak yang terlibat dalam pengujian. Kegiatan ini meliputi:

a. Penentuan tim yang terlibat dalam pengujian, seperti *tester*, *developer*, dan pengguna.
b. Penyusunan skenario pengujian yang akan digunakan selama proses pengujian.
c. Penyiapan dokumen pendukung seperti PDHUPL, dan data uji.

Setelah seluruh persiapan prosedural dilakukan, proses pengujian dapat dilaksanakan sesuai dengan rencana yang telah disusun.

2.5.2.2. Persiapan Perangkat Keras

Subbagian ini menjelaskan proses persiapan perangkat keras yang digunakan dalam pelaksanaan pengujian perangkat lunak. Persiapan perangkat keras dilakukan sebelum pengujian dimulai untuk memastikan seluruh perangkat dapat digunakan dengan baik dan mendukung proses pengujian sistem. Kegiatan yang dilakukan meliputi:

a. Menyiapkan komputer/*laptop* yang akan digunakan sebagai *server* lokal dan *client*.
b. Memastikan seluruh perangkat dalam kondisi baik dan siap digunakan.
c. Menghubungkan perangkat ke jaringan (LAN/WiFi) agar dapat saling terintegrasi.
d. Menyiapkan perangkat tambahan seperti printer untuk pengujian *output* cetak.
e. Memastikan perangkat *input* seperti *keyboard* dan tetikus berfungsi dengan baik.

Perangkat keras yang digunakan beserta spesifikasinya adalah sebagai berikut:

1. **Komputer / *Laptop* (*Server* & Development)**
    a. Prosesor: Intel Core i5 atau setara
    b. RAM: Minimal 8 GB
    c. Penyimpanan: Minimal 256 GB SSD

2. **Komputer / *Laptop* (*Client / Workstation*)**
    a. Jumlah: Minimal 1–2 unit
    b. Prosesor: Intel Core i3 atau setara
    c. RAM: Minimal 4 GB
    d. Penyimpanan: Minimal 128 GB
3. **Perangkat Jaringan**
    a. Router atau hotspot untuk koneksi antar perangkat
    b. Koneksi internet

2.5.2.3. Persiapan Perangkat Lunak

Subbagian ini menjelaskan proses persiapan perangkat lunak yang digunakan dalam pelaksanaan pengujian sistem. Persiapan perangkat lunak dilakukan untuk memastikan seluruh aplikasi dan *tools* pendukung telah terinstal dan dapat berjalan dengan baik sebelum proses pengujian dimulai. Adapun langkah-langkah persiapan perangkat lunak adalah sebagai berikut:

1. Aplikasi sistem (sistem informasi *gym*) disiapkan dalam bentuk file *project* dan ditempatkan pada komputer *server* atau lokal.
2. Instalasi sistem operasi Microsoft Windows 10 64-bit pada perangkat yang digunakan sebagai *server* dan client.
3. Instalasi *web server* dan *database* menggunakan paket seperti XAMPP yang mencakup Apache dan MySQL.
4. Instalasi bahasa pemrograman dan *dependency* yang dibutuhkan, seperti PHP serta *framework* Laravel melalui *Composer*.
5. Konfigurasi *database* dengan mengimpor file *database* (*dump*) ke dalam MySQL sesuai dengan struktur yang telah dirancang.
6. Menjalankan aplikasi melalui *browser* seperti Google Chrome untuk memastikan sistem dapat diakses dan berjalan dengan baik.

2.5.3. Pelaksanaan

Subbagian ini menjelaskan proses pelaksanaan pengujian perangkat lunak yang dilakukan berdasarkan skenario yang telah dirancang. Pelaksanaan pengujian dilakukan untuk setiap fitur atau modul dalam sistem, dengan mengacu pada *use case* yang telah didefinisikan pada dokumen kebutuhan perangkat lunak (SKPL). Pengujian dilakukan dengan pendekatan *black box testing*, yaitu dengan menguji fungsi sistem tanpa melihat kode program.

Setiap pengujian dilakukan berdasarkan:

a. *Use case* yang menggambarkan interaksi antara pengguna dan sistem.
b. *Basic path*, yaitu alur normal ketika sistem berjalan sesuai dengan yang diharapkan.
c. *Alternative path*, yaitu kondisi alternatif atau kemungkinan kesalahan yang dapat terjadi.

Langkah-langkah pelaksanaan pengujian adalah sebagai berikut:

1. Menjalankan sistem sesuai dengan skenario pengujian yang telah dibuat.
2. Memasukkan data uji sesuai dengan *test case*.
3. Mengamati hasil keluaran yang dihasilkan oleh sistem.
4. Mencatat hasil pengujian, apakah sesuai dengan yang diharapkan atau tidak.
5. Mengidentifikasi dan mendokumentasikan kesalahan (*bug*) yang ditemukan.
6. Melakukan pengujian ulang (*retesting*) setelah perbaikan dilakukan oleh *developer*.

Pengujian dilakukan pada seluruh modul utama sistem, seperti:

a. Manajemen pelanggan
b. Manajemen produk
c. *Membership*
d. Jadwal Pelatihan
e. Transaksi produk
f. Transaksi *membership*

Dengan pelaksanaan pengujian ini, diharapkan sistem dapat berjalan sesuai dengan kebutuhan dan bebas dari kesalahan yang dapat mengganggu pengguna.

2.5.4. Pelaporan Hasil

Subbagian ini menjelaskan proses pelaporan hasil pengujian perangkat lunak yang telah dilakukan. Hasil pengujian sistem akan didokumentasikan dalam bentuk laporan yang berisi seluruh aktivitas pengujian, termasuk skenario uji, data uji, hasil yang diharapkan, hasil aktual, serta temuan kesalahan (*bug*) yang ditemukan selama proses pengujian.

Laporan hasil pengujian akan disusun secara sistematis dan disampaikan kepada pihak terkait, yaitu:

a. Pengembang (*developer*), sebagai bahan evaluasi dan perbaikan sistem.
b. Pembimbing atau penguji akademik, sebagai bagian dari penilaian tugas akhir.

Proses pelaporan dilakukan dengan langkah-langkah sebagai berikut:

1. Mengumpulkan seluruh hasil pengujian dari setiap modul sistem.
2. Mendokumentasikan hasil pengujian dalam bentuk tabel atau laporan terstruktur.
3. Mengidentifikasi dan mengelompokkan kesalahan berdasarkan tingkat prioritas.
4. Menyusun kesimpulan terkait kualitas sistem berdasarkan hasil pengujian.
5. Menyampaikan laporan hasil pengujian setelah seluruh proses pengujian selesai dilakukan.

Laporan hasil pengujian ini diharapkan dapat memberikan gambaran mengenai tingkat keberhasilan sistem dalam memenuhi kebutuhan yang telah ditentukan, serta menjadi dasar dalam proses perbaikan dan pengembangan sistem lebih lanjut.

3. Identifikasi dan Rencana Pengujian

3.1.Identifikasi Pengujian

Kelas Uji	Butir Uji	Identifikasi		Tingkat Pengujian	Jenis Pengujian	Jadwal
		SKPL	PDHUPL			
Pengujian *Use Case Login*	*Login* Sistem	UC-001	AU-001	Pengujian Unit	*Black Box*	10/06/2026
	Logout Sistem	UC-001	AU-001-02	Pengujian Unit	*Black Box*	10/06/2026
Pengujian *Use Case* Registrasi	Registrasi pelanggan	UC-002	AU-002	Pengujian Unit	*Black Box*	10/06/2026
Pengujian *Use Case* Kelola *Membership*	Tambah *Membership*	UC-101-01	AU-101-01	Pengujian Unit	*Black Box*	11/06/2026
	Edit Membership	UC-101-02	AU-101-02	Pengujian Unit	*Black Box*	11/06/2026
	Hapus *Membership*	UC-101-03	AU-101-03	Pengujian Unit	*Black Box*	11/06/2026
	Lihat Data *Membership*	UC-101-04	AU-101-04	Pengujian Unit	*Black Box*	11/06/2026
Pengujian *Use Case* Kelola Data Produk	Tambah Produk	UC-201-01	AU-201-01	Pengujian Unit	*Black Box*	12/06/2026
	Edit Produk	UC-201-02	AU-201-02	Pengujian Unit	*Black Box*	12/06/2026
	Hapus Produk	UC-201-03	AU-201-03	Pengujian Unit	*Black Box*	12/06/2026
	Lihat Data Produk	UC-201-04	AU-201-04	Pengujian Unit	*Black Box*	12/06/2026
Pengujian	Hapus Data	UC-501-01	AU-501-	Pengujian	*Black Box*	12/06/20

Use Case Kelola Data pelanggan	Pelanggan		01	Unit		26
Pengujian *Use Case* Kelola Data *Private Trainer*	Tambah *private trainer*	UC-301-01	AU-301-01	Pengujian Unit	*Black Box*	13/06/2026
	Hapus *Private Trainer*	UC-301-03	AU-301-03	Pengujian Unit	*Black Box*	13/06/2026
	Lihat Data *Private Trainer*	UC-301-04	AU-301-04	Pengujian Unit	*Black Box*	13/06/2026
Pengujian Use Case Pembayaran *Membership*	Proses Pembayaran *Membership*	UC-PM-001	PM-001	Pengujian Unit	*Black Box*	14/06/2026
	Validasi Pembayaran *Membership*	UC-PM-002	PM-002	Pengujian Unit	*Black Box*	14/06/2026
	Kirim Bukti Pembayaran *Membership*	UC-PM-003	PM-003	Pengujian Unit	*Black Box*	14/06/2026
	Riwayat Pembayaran *Membership*	UC-PM-004	PM-004	Pengujian Unit	*Black Box*	14/06/2026
Pengujian Use Case	Proses Pembayaran	UC-PP-001	PP-001	Pengujian Unit	*Black Box*	15/06/2026

Pembayaran Produk	Produk					
	Kirim Bukti Pembayaran Produk	UC-PP-002	PP-002	Pengujian Unit	*Black Box*	15/06/2026
	Menunggu Konfirmasi Pembayaran Produk	UC-PP-003	PP-003	Pengujian Unit	*Black Box*	15/06/2026
	Riwayat Pembayaran Produk	UC-PP-004	PP-004	Pengujian Unit	*Black Box*	15/06/2026
Pengujian *Use Case* Laporan	Cetak Laporan *Membership*	UC-601-01	LP-001	Pengujian Integrasi	*Black Box*	15/06/2026
	Cetak Laporan Pembayaran	UC-601-02	LP-002	Pengujian Integrasi	*Black Box*	15/06/2026
	Cetak Laporan Produk	UC-601-03	LP-003	Pengujian Integrasi	*Black Box*	15/06/2026
Pengujian Use Case *Edit* Profil	Menampilkan Data Profil	UC-EP-001	EP-001	Pengujian Unit	*Black Box*	17/06/2026
	Ubah Profil	UC-EP-002	EP-002	Pengujian Unit	*Black Box*	17/06/2026
	Validasi Data *Edit* Profil	UC-EP-003	EP-003	Pengujian Unit	*Black Box*	17/06/2026

	Ubah Password	UC-EP-004	EP-004	Pengujian Unit	*Black Box*	17/06/2026
	Simpan Perubahan Profil	UC-EP-005	EP-005	Pengujian Unit	*Black Box*	17/06/2026
Pengujian Use Case Jadwal Pelatihan	Tambah Jadwal Pelatihan	UC-801-01	AU-801-01	Pengujian Unit	*Black Box*	18/06/2026
	Tampil Data Jadwal Pelatihan	UC-801-02	AU-801-02	Pengujian Unit	*Black Box*	18/06/2026
	Reschedule Jadwal Pelatihan	UC-801-03	AU-801-03	Pengujian Unit	*Black Box*	18/06/2026
	Validasi Bentrok Jadwal Pelatihan	UC-801-04	AU-801-04	Pengujian Unit	*Black Box*	18/06/2026
	Konfirmasi Jadwal Pelatihan	UC-801-05	AU-801-05	Pengujian Unit	*Black Box*	18/06/2026
Pengujian Use Case Informasi *Private Trainer*	Tampil Data Informasi *Private Trainer*	UC-IT-001	IT-001	Pengujian Unit	*Black Box*	19/06/2026
	Tambah	UC-IT-002	IT-002	Pengujian	*Black Box*	19/06/20

	Informasi *Private Trainer*			Unit		26
	Ubah Informasi *Private Trainer*	UC-IT-003	IT-003	Pengujian Unit	*Black Box*	19/06/2026

3.2.Rencana Pengujian

3.2.1. Urutan Pelaksanaan Pengujian

Urutan pelaksanaan pengujian dilakukan sesuai dengan nomor identifikasi pengujian yang telah ditentukan pada Subbab 3.1. Pengujian dimulai dari fitur dasar sistem, kemudian dilanjutkan ke fitur pengelolaan data, transaksi, hingga laporan.

Adapun urutan pelaksanaan pengujian adalah sebagai berikut:

1. Pengujian *Use Case Login*
    a. *Login* Sistem
    b. *Logout* Sistem
2. Pengujian *Use Case* Registrasi
    a. *Registrasi* Pelanggan
3. Pengujian *Use Case* Kelola *Membership*
    a. Tambah *Membership*
    b. *Edit Membership*
    c. Hapus *Membership*
    d. Lihat Data *Membership*
4. Pengujian *Use Case* Kelola Produk

a. Tambah Produk
b. *Edit* Produk
c. Hapus Produk
d. Lihat Data Produk

5. Pengujian *Use Case* Pembayaran *Membership*
    a. Proses Pembayaran *Membership*
    b. Validasi Pembayaran *Membership*
    c. Kirim Bukti Pembayaran
    d. Riwayat Pembayaran *Membership*
6. Pengujian *Use Case* Pembayaran Produk
    a. Proses Pembayaran Produk
    b. Kirim Bukti Pembayaran
    c. Menunggu Konfirmasi Pembayaran
    d. Riwayat Pembayaran Produk
7. Pengujian *Use Case* Kelola Data Pelanggan
    a. Tampil Data pelanggan
    b. Hapus Data pelanggan
8. Pengujian *Use Case* Kelola *Private Trainer*
    a. Tambah Data *Private Trainer*
    b. Hapus data *Private Trainer*
    c. Tampil Data *Private Trainer*
9. Pengujian *Use Case* Laporan
    a. Cetak Laporan *Membership*
    b. Cetak Laporan Pembayaran
    c. Cetak Laporan Produk
10. Pengujian *Use Case* Profil
    a. Tampil Data Profil
    b. *Edit* Data Profil

c. Ubah *Password*
d. Simpan Perubahan Profil

11. Pengujian *Use Case* Jadwal Pelatihan
    a. Tampil Data Jadwal Pelatihan
    b. Tambah Jadwal Pelatihan
    c. Reschedule Data Jadwal Pelatihan
    d. Validasi Bentrok Jadwal Pelatihan
    e. Konfirmasi Jadwal Pelatihan
12. Pengujian *Use case* Informasi *Private Trainer*
    a. Tampil Data Informasi *Private Trainer*
    b. Tambah Data Informasi *Private Trainer*
    c. Ubah Data Informasi *Private Trainer*

3.2.2. Data Pengujian

Data pengujian merupakan data yang digunakan selama proses pengujian sistem untuk memastikan setiap fungsi pada perangkat lunak dapat berjalan sesuai dengan kebutuhan sistem. Data yang digunakan dalam pengujian disesuaikan dengan fitur-fitur yang terdapat pada sistem *membership*.

Adapun data pengujian yang digunakan meliputi:

a. Data pengguna

   Digunakan untuk pengujian proses *login*, registrasi, dan pengelolaan akun pengguna.

b. Data *membership*

   Digunakan untuk pengujian fitur tambah *membership*, *edit membership*, dan hapus *membership*.

c. Data produk

   Digunakan untuk pengujian pengelolaan produk seperti tambah

produk, *edit* produk, hapus produk, pencarian produk, dan tampilan data produk.

d. Data pembayaran *membership*

Digunakan untuk pengujian proses pembayaran *membership*, validasi pembayaran, riwayat pembayaran, dan kirim bukti pembayaran.

e. Data pembayaran produk

Digunakan untuk pengujian proses pembayaran produk, validasi pembayaran, kirim bukti pembayaran, menunggu konfirmasi, dan Riwayat pembayaran.

f. Data *Private Trainer*

Digunakan untuk pengujian pengelolaan *private trainer* seperti tambah *private trainer*, hapus *private trainer*, dan tampilan data *private trainer*.

g. Data laporan

Digunakan untuk pengujian fitur pembuatan dan pencetakan laporan *membership*, laporan pembayaran, laporan perpanjangan *membership*, dan laporan produk.

h. Data Profil

Digunakan untuk pengujian pengelolaan profil *user* seperti tampilan data profil, ubah data profil, ubah *password*, dan simpan perubahan.

i. Data Jadwal Pelatihan

Digunakan untuk pengujian pengelolaan jadwal pelatihan seperti tampilan data jadwal pelatihan, *reschedule* data jadwal pelatihan, validasi bentrok jadwal, Konfirmasi jadwal pelatihan dan tampil jadwal Pelatihan.

j. Data Informasi *Private Trainer*

Digunakan untuk pengujian pengelolaan informasi jadwal pelatihan seperti tampilan data informasi *private trainer*, tambah informasi *private trainer*, dan ubah informasi *private trainer*.

4. Deskripsi Pengujian

4.1 Identifikasi Kelas Pengujian *Use Case Login*

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi *use case Login* dengan aktor sebagai pengguna sistem.

4.1.1 Identifikasi Butir Pengujian *Login* Sistem (AU-001)

Butir pengujian ini melakukan pengujian terhadap antarmuka *login*, dengan *input* berupa *username* dan *password* yang di*input*kan melalui *textbox*.

4.1.2 Identifikasi Butir Pengujian *Logout* Sistem (AU-001-02)

Butir pengujian ini melakukan pengujian terhadap fungsi *logout* sistem untuk memastikan pengguna dapat keluar dari sistem dengan benar dan sesi pengguna berakhir.

4.2 Identifikasi Kelas Pengujian *Use Case* Registrasi

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi registrasi pelanggan dengan aktor pelanggan.

4.2.1 Identifikasi Butir Pengujian Registrasi Pelanggan (AU-002)

Butir pengujian ini melakukan pengujian terhadap antarmuka registrasi pengguna dengan *input* berupa *username*, *password*, nomor telepon, dan alamat.

4.3. Identifikasi Kelas Pengujian Kelola *Membership*

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi pengelolaan *membership* dengan aktor Admin.

4.3.1 Identifikasi Butir Pengujian Tambah *Membership* (AU-101-01)

Butir pengujian ini melakukan pengujian terhadap fungsi tambah *membership* dengan *input* berupa nama *membership*, harga *membership*, durasi *membership*, dan deskripsi *membership*.

4.3.2 Identifikasi Butir Pengujian *Edit Membership* (AU-101-02)

Butir pengujian ini melakukan pengujian terhadap fungsi *edit membership* dengan *input* berupa perubahan data *membership* seperti nama *membership*, harga, dan durasi *membership*.

4.3.3 Identifikasi Butir Pengujian Hapus *Membership* (AU-101-03)

Butir pengujian ini melakukan pengujian terhadap fungsi penghapusan data *membership* dengan *input* berupa data *membership* yang dipilih.

4.3.4 Identifikasi Butir Pengujian Lihat Data *Membership* (AU-101-04)

Butir pengujian ini melakukan pengujian terhadap fungsi menampilkan data *membership* yang tersimpan pada sistem.

4.4 Identifikasi Kelas Pengujian Kelola Produk

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi pengelolaan produk dengan aktor Admin.

4.4.1 Identifikasi Butir Pengujian Tambah Produk (AU-201-01)

Butir pengujian ini melakukan pengujian terhadap fungsi tambah produk dengan *input* berupa nama produk, kategori produk, harga produk, stok produk, dan deskripsi produk.

4.4.2 Identifikasi Butir Pengujian *Edit* Produk (AU-201-02)

Butir pengujian ini melakukan pengujian terhadap fungsi *edit* produk dengan *input* berupa perubahan data produk seperti nama produk, harga, dan stok produk.

4.4.3 Identifikasi Butir Pengujian Hapus Produk (AU-201-03)

Butir pengujian ini melakukan pengujian terhadap fungsi penghapusan produk dengan *input* berupa produk yang dipilih.

4.4.4 Identifikasi Butir Pengujian Lihat Data Produk (AU-201-04)

Butir pengujian ini melakukan pengujian terhadap fungsi menampilkan data produk yang tersimpan pada sistem.

4.5 Identifikasi Kelas Pengujian Kelola Data Pelanggan

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi pengelolaan data pelanggan dengan aktor Admin.

4.5.1 Identifikasi Butir Pengujian Hapus Data Pelanggan (AU-501-03)

Butir pengujian ini melakukan pengujian terhadap fungsi penghapusan data pelanggan dengan *input* berupa pelanggan yang dipilih.

4.6 Identifikasi Kelas Pengujian Kelola *Private Trainer*

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi pengelolaan data *private trainer* dengan aktor Admin sebagai pengguna sistem.

4.6.1 Identifikasi Butir Pengujian Tambah *Private Trainer* (AU-701-01)

Butir pengujian ini melakukan pengujian terhadap fungsi tambah data *private trainer* dengan *input* berupa nama *private trainer*, jenis kelamin, nomor telepon, dan spesialisasi latihan.

4.6.2 Identifikasi Butir Pengujian Hapus *Private Trainer* (AU-701-03)

Butir pengujian ini melakukan pengujian terhadap fungsi penghapusan data *private trainer* dengan *input* berupa data *private trainer* yang dipilih.

4.6.3 Identifikasi Butir Pengujian Lihat Data *Private Trainer* (AU-701-04)

Butir pengujian ini melakukan pengujian terhadap fungsi menampilkan data *private trainer* yang tersimpan pada sistem.

4.7 Identifikasi Kelas Pengujian Pembayaran *Membership*

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi *use case* Pembayaran *Membership* dengan aktor pelanggan dan Admin sebagai pengguna sistem.

4.7.1 Identifikasi Butir Pengujian Proses Pembayaran *Membership* (PM-001)

Butir pengujian ini melakukan pengujian terhadap proses pembayaran *membership*

yang dilakukan oleh pelanggan, dengan *input* berupa pemilihan paket *membership*, metode pembayaran, dan data transaksi yang diinputkan melalui sistem.

4.7.2 Identifikasi Butir Pengujian Validasi Pembayaran *Membership* (PM-002)

Butir pengujian ini melakukan pengujian terhadap fungsi validasi pembayaran *membership* untuk memastikan sistem dapat memeriksa kelengkapan data pembayaran, nominal transaksi, serta status pembayaran sebelum *membership* diaktifkan.

4.7.3 Identifikasi Butir Pengujian Kirim Bukti Pembayaran *Membership* (PM-003)

Butir pengujian ini melakukan pengujian terhadap fungsi unggah bukti pembayaran *membership*, dengan *input* berupa file bukti pembayaran yang dikirimkan pelanggan melalui sistem untuk memastikan file berhasil tersimpan dan dapat diverifikasi.

4.7.4 Identifikasi Butir Pengujian Riwayat Pembayaran *Membership* (PM-004)

Butir pengujian ini melakukan pengujian terhadap tampilan riwayat pembayaran *membership* untuk memastikan sistem dapat menampilkan daftar transaksi pembayaran yang pernah dilakukan pelanggan dengan informasi yang sesuai.

4.8 Identifikasi Kelas Pengujian Pembayaran Produk

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi *use case* Pembayaran Produk dengan aktor pelanggan dan Admin sebagai pengguna sistem.

4.8.1 Identifikasi Butir Pengujian Proses Pembayaran Produk (PP-001)

Butir pengujian ini melakukan pengujian terhadap proses pembayaran produk yang dilakukan oleh pelanggan, dengan *input* berupa data produk yang dipilih, jumlah pembelian, dan metode pembayaran yang diinputkan melalui sistem.

4.8.2 Identifikasi Butir Pengujian Kirim Bukti Pembayaran Produk (PP-002)

Butir pengujian ini melakukan pengujian terhadap fungsi unggah bukti pembayaran produk, dengan *input* berupa file bukti pembayaran yang dikirimkan pelanggan untuk memastikan data pembayaran tersimpan dengan benar pada sistem.

4.8.3 Identifikasi Butir Pengujian Menunggu Konfirmasi Pembayaran Produk (PP-003)

Butir pengujian ini melakukan pengujian terhadap status transaksi pembayaran produk untuk memastikan sistem menampilkan status menunggu konfirmasi pembayaran setelah pelanggan berhasil mengirim bukti pembayaran dan sebelum diverifikasi oleh admin.

4.8.4 Identifikasi Butir Pengujian Riwayat Pembayaran Produk (PP-004)

Butir pengujian ini melakukan pengujian terhadap tampilan riwayat pembayaran produk untuk memastikan sistem dapat menampilkan data transaksi produk yang telah dilakukan pelanggan secara lengkap dan sesuai.

4.9 Identifikasi Kelas Pengujian Laporan

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi laporan sistem dengan aktor Admin.

4.9.1 Identifikasi Butir Pengujian Menampilkan Laporan (LP-001)

Butir pengujian ini melakukan pengujian terhadap tampilan laporan untuk memastikan sistem dapat menampilkan data transaksi produk dan *membership* sesuai data yang tersimpan.

4.9.2 Identifikasi Butir Pengujian Filter Laporan (LP-002)

Butir pengujian ini melakukan pengujian terhadap fungsi filter laporan berdasarkan periode tertentu untuk memastikan data yang ditampilkan sesuai rentang tanggal yang dipilih.

4.9.3 Identifikasi Butir Pengujian Cetak Laporan (LP-003)

Butir pengujian ini melakukan pengujian terhadap fungsi cetak laporan untuk memastikan laporan dapat dicetak atau diekspor dengan benar.

4.10 Identifikasi Kelas Pengujian *Use Case Edit* Profil

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi *use case Edit*

Profil dengan aktor pelanggan sebagai pengguna sistem.

4.10.1 Identifikasi Butir Pengujian Menampilkan Data Profil (EP-001)

Butir pengujian ini melakukan pengujian terhadap tampilan data profil pengguna untuk memastikan sistem dapat menampilkan data sesuai informasi yang tersimpan pada *database*.

4.10.2 Identifikasi Butir Pengujian *Edit* Profil (EP-002)

Butir pengujian ini melakukan pengujian terhadap fungsi *edit* profil, dengan *input* berupa perubahan data pengguna melalui *form* yang tersedia pada sistem.

4.10.3 Identifikasi Butir Pengujian Validasi Data *Edit* Profil (EP-003)

Butir pengujian ini melakukan pengujian terhadap validasi data *edit* profil untuk memastikan sistem dapat mendeteksi kesalahan *input* dan memberikan pesan validasi.

4.10.4 Identifikasi Butir Pengujian Ubah *Password* (EP-004)

Butir pengujian ini melakukan pengujian terhadap fungsi ubah *password* untuk memastikan sistem dapat memverifikasi *password* lama, menerima *password* baru yang valid, serta menyimpan perubahan *password* dengan benar.

4.10.5 Identifikasi Butir Pengujian Simpan Perubahan Profil (EP-005)

Butir pengujian ini melakukan pengujian terhadap fungsi penyimpanan perubahan profil untuk memastikan data profil dan perubahan *password* berhasil diperbarui pada sistem.

4.11 Identifikasi Kelas Pengujian *Use Case* Jadwal Pelatihan

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi pengelolaan jadwal pelatihan dengan aktor *Private Trainer* dan pelanggan sebagai pengguna sistem.

4.11.1 Identifikasi Butir Pengujian Tambah Jadwal Pelatihan (AU-801-01)

Butir pengujian ini melakukan pengujian terhadap fungsi penambahan jadwal pelatihan dengan *input* berupa tanggal, jam mulai dan jam selesai.

4.11.2 Identifikasi Butir Pengujian Tampil Data Jadwal Pelatihan (AU-801-02)

Butir pengujian ini melakukan pengujian terhadap fungsi tampilan data jadwal pelatihan yang tersimpan pada sistem.

4.11.3 Identifikasi Butir Pengujian Reschedule Jadwal Pelatihan (AU-801-03)

Butir pengujian ini melakukan pengujian terhadap fungsi perubahan jadwal (*reschedule*) pelatihan dengan *input* berupa tanggal dan waktu pelatihan baru.

4.11.4 Identifikasi Butir Pengujian Validasi Bentrok Jadwal Pelatihan (AU-801-04)

Butir pengujian ini melakukan pengujian terhadap validasi sistem untuk memastikan tidak terjadi bentrok jadwal antara pelanggan dan *private trainer*.

4.11.5 Identifikasi Butir Pengujian Konfirmasi Jadwal Pelatihan (AU-801-05)

Butir pengujian ini melakukan pengujian terhadap status jadwal pelatihan untuk memastikan sistem menampilkan status pending setelah pelanggan berhasil membuat jadwal pelatihan dan sebelum diverifikasi oleh *private trainer*.

4.12 Identifikasi Kelas Pengujian Informasi *Private Trainer*

Kelas pengujian ini meliputi pengujian-pengujian yang melibatkan fungsi pengelolaan informasi *private trainer* dengan aktor *Private Trainer* sebagai pengguna sistem.

4.12.1 Identifikasi Butir Pengujian Tampil Data *Private Trainer* (IT-001)

Pengujian ini dilakukan untuk memastikan sistem dapat menampilkan seluruh data *private trainer* yang tersimpan dalam *database* secara lengkap, akurat, dan sesuai dengan hak akses pengguna.

4.12.2 Pengujian Tambah Data *Private Trainer* (IT-002)

Pengujian ini dilakukan untuk memastikan informasi *private trainer* telah diisi dan sesuai. Pengujian mencakup validasi data masukan, proses penyimpanan data, dan notifikasi keberhasilan atau kegagalan penyimpanan.

4.12.3 Pengujian Ubah Data *Private Trainer* (IT-003)

Pengujian ini dilakukan untuk memastikan *private trainer* dapat mengubah data yang

telah tersimpan dalam sistem. Pengujian mencakup validasi data yang diubah, proses pembaruan data, dan notifikasi hasil perubahan data.

5. Hasil Pengujian

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
AU-001_01	Pengujian *Use Case Login*	5. Buka *form login*. 6. Masukkan *username* valid. 7. Entri *password* valid 8. Tekan tombol '*Login*'	*Username* = 'admin' *Password* = 'admin'	Pesan: "user masuk ke aplikasi"	User masuk ke dashboard sesuai dengan *role*.	User masuk ke dashboard sesuai dengan *role*.	Handal
AU-001_02	Pengujian *Use Case Login*	5. Buka *form login*. 6. Entri *username* invalid.	*Username* = 'admine' *Password* = 'admin'	Pesan: "*username*/*password* tidak salah"	Kelar pesan: "*username*/*password* tidak valid"	Kelar pesan: "*username*/*password* tidak valid"	Handal

		7. Entri sembarang *password* 8. Tekan tombol '*Login*'					
AU-001_03	Pengujian *Use Case Login*	5. Buka *form login*. 6. Entri sembarang *username*. 7. Entri invalid *password* 8. Tekan tombol '*Login*'	*Username* = 'admin' *Password* = 'admine'	Pesan: "*username/password* tidak valid"	Kelau pesan: "*username/password* tidak valid"	Kelau pesan: "*username/password* tidak valid"	Tidak Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
AU-002_01	Pengujian *Use Case Register*	1. Buka *form Register* 2. Masukan *username* valid. 3. Masukan *password* valid 4. Masukan Nomor Telepon valid 5. Masukan Alamat valid	*Username* = 'Bulan' *Password* = '111111' Nomor Telepon = "086473846372" Alamat = "Jl. Kekiri Kekanan No 99"	Pesan: "*Register* Berhasil"	Keluar Pesan: "*Register* Berhasil"	User langsung diarahkan ke halaman *login*	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		6. Tekan tombol '*Register*'					
AU-002_02	Pengujian *Use Case Register*	1. Buka *form Register* 2. Masukan *username* yang sudah ada. 3. Masukan *password* valid. 4. Masukan Nomor Telepon valid.	*Username* = 'Bulan' *Password* = '111111' Nomor Telepon = "086473846372" Alamat = "Jl. Kekiri Kekanan No 99"	Pesan: "*Username* sudah ada"	Keluar Pesan: "*Username* sudah ada"	Keluar Pesan: "*Username* sudah ada"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		5. Masukan Alamat valid. 6. Tekan tombol '*Register*'					
AU-002_03	Pengujian *Use Case Register*	1. Buka *form Register* 2. Masukan *username* valid. 3. Masukan *password* kurang atau lebih dari 6 digit. 4. Masukan Nomor	*Username* = 'Bulan' *Password* = '1111111' Nomor Telepon = "086473846372" Alamat = "Jl. Kekiri Kekanan No 99"	Pesan: "*Password* invalid"	Pesan: "*Password* invalid"	Pesan: "*Password* invalid"	Kurang Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		Telepon valid. 5. Masukan Alamat valid. 6. Tekan tombol '*Register*'					
AU-002_02	Pengujian *Use Case Register*	1. Buka *form Register* 2. Masukan *username* valid. 3. Masukan *password* valid.	*Username* = 'Bulan' *Password* = '111111' Nomor Telepon = "08647384637" Alamat = "Jl. Kekiri Kekanan	Pesan: "Nomor telepon Invalid"	Pesan: "Nomor telepon Invalid"	Pesan: "Nomor telepon Invalid"	Kurang Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		4. Masukan Nomor Telepon lebih dari 12 digit. 5. Masukan Alamat valid. 6. Tekan tombol '*Register*'	No 99"				

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
AU-111_01	Pengujian Tambah	1. Buka menu *Membership*.	Tipe = '1 Bulan' Kategori	Pesan: "*Membership*	Keluar Pesan: "*Membership*	Keluar Pesan: "*Membership*	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
	Membership	2. Tekan tombol tambah *membership.* 3. Isi data *membership*. 4. Tekan tombol "Simpan"	'Bundle' Durasi = '1 Bulan' Sesi = '12 sesi' Male = '250.000' Female = '250.000' Couple = '600.000'	Berhasil Ditambahkan"	Berhasil Ditambahkan"	Berhasil Ditambahkan"	
AU-111_02	Pengujian *Edit Membership*	1. Buka menu *Membership*. 2. Tekan tombol *edit membership.* 3. Isi data *membership* yang ingin diubah.	Tipe = '1 Bulan' Kategori 'Reguler' Durasi = '3 Bulan' Male = '250.000' Female = '250.000' Couple =	Pesan: "*Membership* Berhasil Diubah"	Pesan: "*Membership* Berhasil Diubah"	Pesan: "*Membership* Berhasil Diubah"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		4. Tekan tombol "Simpan"	'600.000'				
AU-111_03	Pengujian Hapus *Membership*	1. Buka menu *Membership*. 2. Tekan tombol Hapus. 3. Tekan tombol "Ya" untuk konfirmasi.	Tipe = '1 Bulan' Kategori 'Bundle' Durasi = '1 Bulan' Sesi = '12 sesi' Male = '250.000' Female = '250.000' Couple = '600.000'	Pesan: "*Membership* Berhasil Dihapus"	Pesan: "*Membership* Berhasil Dihapus"	Pesan: "*Membership* Berhasil Dihapus"	Handal
AU-121_01	Pengujian Tambah Produk	1. Buka menu Produk.	Foto = 'isi foto produk' Nama = ' Whey Protein'	Pesan: "Produk Berhasil Ditambahkan"	Pesan: "Produk Berhasil Ditambahkan"	Pesan: "Produk Berhasil Ditambahkan"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		2. Tekan tombol tambah produk. 3. Isi data produk. 4. Tekan tombol "Simpan".	Deskripsi = 'Susu Protein penambah berat badan' Harga = '1.500.000' Stok = '20'				
AU-121_02	Pengujian *Edit* Produk	1. Buka menu Produk. 2. Tekan tombol *Edit*. 3. Isi data produk yang ingin diubah. 4. Tekan tombol "Simpan".	Foto = 'isi foto produk' Nama = ' Whey Protein' Deskripsi = 'Susu Protein penambah berat badan' Harga = '1.500.000'	Pesan: "Produk Berhasil Diubah"	Pesan: "Produk Berhasil Diubah"	Pesan: "Produk Berhasil Diubah"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
			Stok = ’17’				
AU-121_03	Pengujian Hapus Produk	1. Buka menu *Membership*. 2. Tekan tombol Hapus. 3. Tekan tombol “Ya” untuk konfirmasi.	Foto = ’isi foto produk’ Nama = ’ Whey Protein’ Deskripsi = ‘Susu Protein penambah berat badan’ Harga = ’1.500.000’ Stok = ’20’ Couple = ‘600.000’	Pesan: “Produk Berhasil Dihapus”	Pesan: “Produk Berhasil Dihapus”	Pesan: “ Berhasil Dihapus”	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
AU-131_01	Pengujian Tambah *User Private Trainer*	1. Buka menu *Private Trainer*. 2. Tekan tombol tambah *Private Trainer*. 3. Isi data *user private trainer*. 4. Tekan tombol "Simpan"	*Username* = 'Asep' *Password* = '111111' No Telepon = '087648374623 Alamat = 'Jl. Anggrek No. 99'	Pesan: "*Private Trainer* Berhasil Ditambahkan"	Keluar Pesan: "*Private Trainer* Berhasil Ditambahkan"	Keluar Pesan: "*Private Trainer* Berhasil Ditambahkan"	Handal
AU-131_03	Pengujian Hapus *Private Trainer*	1. Buka menu *Private Trainer*. 2. Tekan tombol Hapus.	*Username* = 'Asep' *Password* = '111111' No Telepon = '087648374623	Pesan: "*Private Trainer* Berhasil Dihapus"	Pesan: "*Private Trainer* Berhasil Dihapus"	Pesan: "*Private Trainer* Berhasil Dihapus"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		3. Tekan tombol "Ya" untuk konfirmasi.	Alamat = 'Jl. Anggrek No. 99'				
AU-131_04	Pengujian Tampil data *private trainer*	1. Login dengan *user* admin 2. Buka menu **Kelola *Private Trainer***	Data *Private Trainer*	Sistem menampilkan daftar *private trainer*	Sistem menampilkan daftar *private trainer*	Sistem menampilkan daftar *private trainer*	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
AU-141_01	Pengujian Hapus Data Pelanggan	3. Buka menu data Pelanggan. 4. Tekan tombol Hapus.	*Username* = 'Bulan' *Password* = '111111' No Telepon = '087648374623	Pesan: "Pelanggan Berhasil Dihapus"	Keluar Pesan: "Pelanggan Berhasil Dihapus"	Keluar Pesan: "Pelanggan Berhasil Dihapus"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		5. Tekan tombol "iya" untuk konfirmasi	Alamat = 'Jl. Kekanan Kekiri no 99'				

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
PM-001_01	Pengujian Proses Pembayaran *Membership*	1. *Login* ke sistem. 2. Buka menu *Membership*. 3. Pilih paket *membership*. 4. Tekan tombol **Daftar Sekarang.** 5. Pilih metode pembayaran.	Nama *Membership* = '1 Bulan' Harga = 'Rp250.000' Metode Pembayaran = 'Transfer Bank'	Pesan : "Pembayaran *Membership* Berhasil Diproses"	Keluar pesan: "Pembayaran *Membership* Berhasil Diproses"	Keluar pesan: "Pembayaran *Membership* Berhasil Diproses"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		6. Tekan tombol **Bayar Sekarang**.					
PM-001_02	Pengujian Validasi Pembayaran *Membership*	1. *Login* ke sistem. 2. Buka menu *Membership*. 3. Pilih paket *membership*. 4. Tekan tombol **Daftar Sekarang** 5. Kosongkan metode pembayaran. 6. Tekan tombol **Bayar Sekarang**.	Nama *Membership* = '1 Bulan' Harga = 'Rp250.000' Metode Pembayaran = '(kosong)'	Pesan: "Metode pembayaran wajib diisi"	Keluar pesan validasi sesuai *input* kosong	Pesan: "Metode pembayaran wajib diisi"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
PM-001_03	Pengujian Kirim Bukti Pembayaran *Membership*	1. *Login* ke sistem. 2. Buka menu Riwayat *Membership*. 3. Pilih transaksi *membership*. 4. Tekan tombol **Upload Bukti Pembayara** 5. Pilih file bukti pembayaran. 6. Tekan tombol **Bayar Sekarang**.	File = 'bukti_transfer.jpg'	Pesan: "Pembayaran *Membership* Berhasil Diproses"	Pesan: "Pembayaran *Membership* Berhasil Diproses"	Pesan: "Pembayaran *Membership* Berhasil Diproses"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
PM-001_04	Pengujian Riwayat Pembayaran *Membership*	1. *Login* ke sistem. 2. Buka menu **Riwayat Transaksi**.	Data transaksi *membership*	Sistem menampilkan daftar riwayat pembayaran *membership*	Sistem menampilkan daftar riwayat pembayaran *membership*	Sistem menampilkan daftar riwayat pembayaran *membership*	Handal
PM-001_05	Pengujian Konfirmasi Pembayaran *Membership*	1. *Login* sebagai admin. 2. Buka menu Kelola Transaksi *Membership*. 3. Pilih transaksi *membership* yang ingin dikonfirmasi. 4. Tekan tombol **Approve**.	Status = 'Menunggu Konfirmasi'	Pesan: "Pembayaran *Membership* Berhasil Dikonfirmasi"	Status pembayaran berubah menjadi berhasil	Pesan berhasil ditampilkan dan status berubah	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
PP-001_01	Pengujian Proses Tambah keranjang	1. *Login* ke sistem. 2. Buka menu **Produk Protein**. 3. Pilih produk yang akan dibeli. 4. Tekan tombol **Tambah ke Keranjang**.	Data Keranjang	Pesan : "Produk berhasil ditambahkan ke keranjang"	Keluar pesan: "Produk berhasil ditambahkan ke keranjang"	Pesan : "Produk berhasil ditambahkan ke keranjang"	Handal
PP-001_02	Pengujian Proses Pembayaran Produk	1. *Login* ke sistem. 2. Buka menu **Produk**. 3. Pilih produk yang akan dibeli.	Nama Produk = 'Whey Protein' Jumlah = '1' Harga = 'Rp800.000' Metode Pembayaran =	Pesan: "Pembayaran Produk Berhasil Diproses"	Keluar pesan: "Pembayaran Produk Berhasil Diproses"	Keluar pesan: "Pembayaran Produk Berhasil Diproses"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		4. Tekan tombol **Tambah ke Keranjang**. 5. Buka menu **Keranjang**. 6. Pilih produk yang ingin dibayar pada keranjang. 7. Tekan tombol **Checkout Sekarang**. 8. Pilih metode pembayaran. 9. Tekan tombol **Bayar Sekarang**.	'Transfer Bank'				

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
PP-001_03	Pengujian Validasi Pembayaran Produk	1. *Login* ke sistem. 2. Buka menu **Produk**. 3. Pilih produk yang akan dibeli. 4. Tekan tombol **Tambah ke Keranjang**. 5. Buka menu **Keranjang**. 6. Pilih produk yang ingin dibayar pada keranjang.	Nama Produk = 'Whey Protein' Jumlah = '1' Harga = 'Rp800.000' Metode Pembayaran = '(kosong)'	Pesan: "Metode pembayaran wajib diisi"	Keluar pesan validasi sesuai *input* kosong	Pesan: "Metode pembayaran wajib diisi"	Kurang Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		7. Tekan tombol **Checkout Sekarang**. 8. Kosongkan metode pembayaran. 9. Tekan tombol **Bayar Sekarang**.					
PP-001_04	Pengujian Kirim Bukti Pembayaran Produk	1. *Login* ke sistem. 2. Buka menu **Produk**. 3. Pilih produk yang akan dibeli.	File = 'bukti_transfer.jpg'	Pesan: "Pembayaran Produk Berhasil Diproses"	Pesan: "Pembayaran Produk Berhasil Diproses"	Pesan: "Pembayaran Produk Berhasil Diproses"	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		4. Tekan tombol **Tambah ke Keranjang**. 5. Buka menu **Keranjang**. 6. Pilih produk yang ingin dibayar pada keranjang. 7. Tekan tombol **Checkout Sekarang**. 8. Pilih metode pembayaran. 9. Upload Bukti Pembayaran					

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		10. Tekan tombol **Bayar Sekarang**.					
PP-001_05	Pengujian Riwayat Pembayaran Produk	1. *Login* ke sistem. 2. Buka menu **Riwayat Transaksi**.	Data transaksi produk	Sistem menampilkan daftar riwayat pembayaran produk	Riwayat transaksi tampil sesuai data	Riwayat pembayaran berhasil ditampilkan	Handal
PM-001_06	Pengujian Konfirmasi Pembayaran *Membership*	1. *Login* sebagai admin. 2. Buka menu Kelola Transaksi Produk. 3. Pilih transaksi *membership* yang ingin dikonfirmasi.	Status = 'Menunggu Konfirmasi'	Pesan: "Pembayaran *Membership* Berhasil Dikonfirmasi"	Status pembayaran berubah menjadi berhasil	Pesan berhasil ditampilkan dan status berubah	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		4. Tekan tombol ***Approve***.					

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
AU-801-01	Pengujian Tambah Jadwal Pelatihan	5. Buka menu jadwal pelatihan. 6. Tekan tombol tambah jadwal. 7. Isi data jadwal. 8. Tekan tombol simpan.	Tanggal = '14 Mei 2026' Jam Mulai = '11.00' Jam Selesai = '13.00'	Pesan "Menunggu Verifikasi *Private trainer*".	Keluar pesan: "Menunggu Verifikasi *Private trainer*".	Pesan : "Menunggu Verifikasi *Private trainer*".	Handal
AU-801-02	Pengujian Tampil Jadwal	3. Buka menu jadwal pelatihan.	Data jadwal pelatihan	Data jadwal pelatihan tampil pada sistem.	Sistem berhasil menampilkan seluruh data	Data jadwal tampil dengan benar.	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
	Pelatihan	4. Pilih tampil data jadwal.			jadwal.		
AU-801-03	Pengujian *Reschedule* Jadwal Pelatihan	4. Pilih jadwal pelatihan. 5. Pilih tanggal dan jam baru. 6. Tekan reschedule.	Tanggal Baru = '12/06/2026' Jam Mulai Baru = '10:00' Jam Selesai Baru = '12:00'	Pesan "Menunggu Verifikasi *Private trainer*".	Pesan "Menunggu Verifikasi *Private trainer*".	Pesan "Menunggu Verifikasi *Private trainer*".	Handal
AU-801-04	Pengujian Validasi Bentrok Jadwal	3. Tambahkan jadwal pada waktu yang sama. 4. Tekan simpan.	Tanggal = '14 Mei 2026' Jam Mulai = '10.00' Jam Selesai = '10.00'	Pesan "Jadwal bentrok dengan jadwal lain".	Keluar Pesan "Jadwal bentrok dengan jadwal lain".	Pesan "Jadwal bentrok dengan jadwal lain".	Handal
AU-801-05	Konfirmasi Reschedule Jadwal Pelatihan	5. Buka menu jadwal pelatihan.	Status *Pending*	Pesan "Jadwal pelatihan berhasil dikonfirmasi"	Pesan "Jadwal pelatihan berhasil dikonfirmasi" tampil dan status	Pesan "Jadwal pelatihan berhasil dikonfirmasi" tampil dan status	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		6. Pilih jadwal yang tersedia. 7. Tekan tombol "Acc". 8. Simpan perubahan status jadwal.		tampil dan status jadwal berubah menjadi "*Approved*".	jadwal berubah menjadi "*Approved*".	jadwal berubah menjadi "*Approved*".	

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
EP-001_01	Pengujian Menampilkan Data Profil	3. *Login* ke sistem. 4. Buka menu **Profil**.	Data profil pengguna	Sistem menampilkan informasi profil pengguna	Data profil tampil sesuai data pada *database*	Data profil berhasil ditampilkan	Handal
EP-001_02	Pengujian *Edit* Profil	6. *Login* ke sistem. 7. Buka menu **Profil**.	Nama = 'Soleh No Telepon = '087648374623' Alamat = 'Jl.	Pesan: **"Profil Berhasil Diperbarui"**	Keluar pesan: **"Profil Berhasil Diperbarui"**seluruh data jadwal.	Keluar pesan: **"Profil Berhasil Diperbarui"**	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		8. Tekan tombol ***Edit* Profil**. 9. Ubah data profil. 10. Tekan tombol **Simpan Perubahan**.	Anggrek No. 99’				
EP-001_03	Pengujian Validasi Data *Edit* Profil	6. *Login* ke sistem. 7. Buka menu **Profil**. 8. Tekan tombol ***Edit* Profil**. 9. Kosongkan data wajib. 10. Tekan tombol **Simpan Perubahan**.	Nama = ‘’ (kosong) No Telepon = ‘’ (kosong)	Pesan: “Data wajib diisi”	Keluar Pesan: “Data wajib diisi”	Keluar Pesan: “Data wajib diisi”	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
EP-001_04	Pengujian Ubah *Password*	6. *Login* ke sistem. 7. Buka menu **Profil**. 8. Pilih menu **Ubah *Password***. 9. Isi *password* lama, *password* baru, dan konfirmasi *password*. 10. Tekan tombol **Update *Password***.	*Password* lama = '111111' *Password* baru = 'qqqqqq' Konfirmasi *Password* = 'qqqqqq'	Pesan: "*Password* Berhasil Diubah"	*Password* berhasil diperbarui dan dapat digunakan *login* kembali	Keluar Pesan: "*Password* Berhasil Diubah"	Handal
EP-001_05	Pengujian Simpan	5. *Login* ke sistem.	Data profil baru	Sistem menyimpan dan	Data berhasil tersimpan pada	Data berhasil diperbarui	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
	Perubahan Profil	6. *Edit* data profil. 7. Tekan tombol **Simpan**. 8. Refresh halaman profil.		menampilkan data terbaru	sistem		

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
LP-001_01	Pengujian Menampilkan Laporan Transaksi	1. *Login* sebagai admin. 2. Buka menu ***Dashboard***. 3. Pilih jenis laporan transaksi.	Data transaksi produk dan *membership*	Sistem menampilkan data laporan transaksi	Data laporan tampil sesuai *database*	Data laporan berhasil ditampilkan	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
LP-001_02	Pengujian Filter Laporan Berdasarkan Tanggal	1. *Login* sebagai admin. 2. Buka menu ***Dashboard***. 3. Pilih rentang tanggal. 4. Tekan tombol **Filter**.	Tanggal Awal = '01-01-2026' Tanggal Akhir = '31-01-2026'	Sistem menampilkan laporan sesuai rentang tanggal	Data laporan sesuai filter tanggal	Data laporan berhasil difilter	Handal
LP-001_03	Pengujian Cetak Laporan	1. *Login* sebagai admin. 2. Buka menu **Dashboard**. 3. Pilih periode laporan. 4. Tekan tombol **Download PDF**.	Periode = 'Januari 2026'	Sistem menghasilkan file atau tampilan print laporan	Laporan dapat dicetak tanpa eror	Laporan berhasil dicetak	Handal

Identifikasi	**Deskripsi**	**Prosedur Pengujian**	**Masukan**	**Keluaran yang diharapkan**	**Kriteria Evaluasi Hasil**	**Hasil yang Didapat**	**Kesimpulan**
IT-001	Pengujian Menampilkan Informasi *Private Trainer*	1. *Login* sebagai *private trainer*. 2. Buka menu ***Biodata***.	Data informasi *Private Trainer*	Sistem menampilkan data informasi *private trainer*	Data *private trainer* tampil sesuai *database*	Data *private trainer* berhasil ditampilkan	Handal
IT-002	Pengujian Tambah Informasi *Private Trainer*	1. *Login* sebagai *private trainer*. 2. Buka menu **Biodata**. 3. Tekan tombol Tambah. 4. Isi data informasi. 5. Tekan tombol Simpan.	Nama = "Asep" Spesialis = "Body Builder" Tanggal Lahir = “20 April 1999” Jenis Kelamin = “Laki - Laki” Pengalaman = “Sertifikasi Body Builder” Foto = “Foto.jpg”	Sistem menampilkan pesan **"Informasi berhasil ditambahkan."**	Data tersimpan pada *database* dan pesan berhasil ditampilkan.	Pesan "Informasi berhasil ditambahkan." muncul dan data tersimpan	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
IT-002_1	Pengujian Tambah Informasi *Private Trainer*	1. *Login* sebagai *private trainer*. 2. Buka menu **Biodata**. 3. Tekan tombol Tambah. 4. Isi data informasi. 5. Tekan tombol Simpan.	Nama = "Asep" Spesialis = "" Tanggal Lahir = “20 April 1999” Jenis Kelamin = “Laki - Laki” Pengalaman = “Sertifikasi Body Builder” Foto = “Foto.jpg”	Sistem menampilkan pesan **"Data harus diisi semua."**	Sistem keluar pesan **"Data harus diisi semua."**	Sistem menampilkan pesan **"Data harus diisi semua."**	Kurang Handal
IT-003	Pengujian Ubah Informasi *Private Trainer*	1. Login sebagai *private trainer*. 2. Buka menu **Biodata**. 3. Tekan tombol *Edit*.	Nama = "Asep D" Spesialis = "Body Builder" Tanggal Lahir = “20 April 1999” Jenis Kelamin =	Sistem menampilkan pesan **"Informasi berhasil diperbarui."**	Perubahan data tersimpan dan pesan berhasil ditampilkan.	Pesan "Informasi berhasil diperbarui." muncul dan data berhasil diperbarui.	Handal

Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Keluaran yang diharapkan	Kriteria Evaluasi Hasil	Hasil yang Didapat	Kesimpulan
		4. Ubah data informasi. 5. Tekan tombol Simpan.	"Laki - Laki" Pengalaman = "Sertifikasi Body Builder" Foto = "Foto.jpg"				

Lampiran Hasil Kuesioner

Timestamp	Nama	Usia	Jenis Kelamin	Sistem informasi gym memudahkan pengelolaan d	Apakah sistem membantu mempercepat proses ad	Apakah sistem memudahkan pengelolaan jadwal l	Apakah sistem membantu mengurangi kesalahan	Apakah sistem memudahkan pencatatan dan peng	Apakah sistem membuat pengelolaan operasional	Tuliskan saran atau masukan Anda untuk pengemb
17/06/2026 21:26:19	Kevin Wilson	20–30 Tahun	Laki - Laki	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Sangat Setuju (5)	Semoga bisa dibuat aplikasi agar lebih gampang di
18/06/2026 18:31:20	Kristoforus Juansandi Yi	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	
18/06/2026 18:33:44	Nugrohanto K	> 40 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Bisa lebih dikembangkan
18/06/2026 18:38:29	Geva	< 20 Tahun	Perempuan	Setuju (4)	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Sangat Setuju (5)	Setuju (4)	
18/06/2026 18:38:50	Nethania Kristi	20–30 Tahun	Perempuan	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	
18/06/2026 18:46:02	Galuh Kusuma Putri	> 40 Tahun	Perempuan	Setuju (4)	Setuju (4)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Setuju (4)	UI dibuat lebih menarik dan mudah dipahami
18/06/2026 18:47:06	Angelice Chilly	20–30 Tahun	Perempuan	Setuju (4)	Setuju (4)	Setuju (4)	Netral (3)	Setuju (4)	Setuju (4)	Bisa menambahkan jadwal baik dari appointment a
18/06/2026 18:50:28	Cia Wicaksono	> 40 Tahun	Perempuan	Setuju (4)	Setuju (4)	Sangat Setuju (5)	Setuju (4)	Netral (3)	Netral (3)	Buat aplikasi android dan iphone juga
18/06/2026 18:51:58	David	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Netral (3)	Setuju (4)	
18/06/2026 18:57:55	Titus	20–30 Tahun	Laki - Laki	Setuju (4)	Setuju (4)	Netral (3)	Setuju (4)	Setuju (4)	Netral (3)	Kalau bisa ada notifikasi yang disambungkan ke W
18/06/2026 19:21:05	Pindo	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Netral (3)	Setuju (4)	Setuju (4)	Sangat Setuju (5)	Untuk pengembangan sistem informasi gym berba
18/06/2026 19:34:23	Ellen	20–30 Tahun	Perempuan	Netral (3)	Netral (3)	Setuju (4)	Netral (3)	Netral (3)	Netral (3)	
18/06/2026 20:03:19	Bernardino Talu Lian	20–30 Tahun	Laki - Laki	Setuju (4)	Netral (3)	Setuju (4)	Setuju (4)	Netral (3)	Setuju (4)	
18/06/2026 20:07:48	Kidung Kinanthi	20–30 Tahun	Perempuan	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Sangat Setuju (5)	Menambah notifikasi otomatis melalui email terkai
18/06/2026 20:20:49	Andrea Christian Pirlo	20–30 Tahun	Laki - Laki	Setuju (4)	Setuju (4)	Setuju (4)	Setuju (4)	Setuju (4)	Setuju (4)	Terus kembangkan
18/06/2026 20:24:03	Stefanus Audy Advent Kr	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Berikan elemen elemen UI agar lebih menarik
18/06/2026 20:58:23	Gede Pandu Prayaksa	20–30 Tahun	Laki - Laki	Setuju (4)	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Setuju (4)	Setuju (4)	
18/06/2026 21:19:11	Antin	> 40 Tahun	Perempuan	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Untuk mempermudah sistem
18/06/2026 21:27:38	Naufal	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Setuju (4)	Netral (3)	Setuju (4)	Kasih video tutorial gym bagi pemula
18/06/2026 21:42:02	Yulianto Puguh	20–30 Tahun	Laki - Laki	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Perlu dilakukan maaa percobaan untuk menganalis
18/06/2026 21:54:10	kevin putra	< 20 Tahun	Laki - Laki	Setuju (4)	Netral (3)	Netral (3)	Netral (3)	Netral (3)	Setuju (4)	Fokus selesaikan fitur utama terlebih dahulu (MVP
18/06/2026 22:01:31	Gatot Subroto	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Dari segi konsep woke banget ndan. Jadi gausah d
18/06/2026 23:03:14	DuBistGutGenug	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Setuju (4)	Netral (3)	Setuju (4)	
18/06/2026 23:44:30	adrian	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sambung & konekaikan dengan aplikasi berbasis k
19/06/2026 1:14:24	Riski Fahmi	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Netral (3)	persimpel halaman awal agar lebih minimaliks
19/06/2026 1:19:34	AXEL ABISMANYU	< 20 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	
19/06/2026 1:20:28	Parjo supratman	< 20 Tahun	Laki - Laki	Setuju (4)	Setuju (4)	Setuju (4)	Setuju (4)	Setuju (4)	Setuju (4)	Gas boleh biar mudah daftar gym
19/06/2026 8:27:30	Nurul Afif Hanifah	20–30 Tahun	Perempuan	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Untuk masukannya di *website* tersebut harap berik
19/06/2026 9:07:22	Sindy	20–30 Tahun	Perempuan	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	
19/06/2026 9:45:16	Enjelina Cristiana Umpan	20–30 Tahun	Perempuan	Netral (3)	Netral (3)	Netral (3)	Netral (3)	Netral (3)	Netral (3)	Fokuslah pada pengembangan Sistem Booking Kel
19/06/2026 12:47:45	Lea Stephani	31–40 Tahun	Perempuan	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Sistem datanya dibuat update sehingga bisa memu
19/06/2026 12:47:52	Allimatu Sadiyah	20–30 Tahun	Perempuan	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Adanya menu tentang pemberitahuan penggunaan
19/06/2026 12:48:05	Vilily	20–30 Tahun	Perempuan	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Tambahkan menu pilihan alat yang digunakan aesu
19/06/2026 13:19:08	Okto Melano	> 40 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	
19/06/2026 19:43:05	Prawoto Agus Santoso	> 40 Tahun	Laki - Laki	Setuju (4)	Setuju (4)	Setuju (4)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Pengkinian data minimal 1 bulan sekali8
19/06/2026 22:15:11	juvenus	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Sangat Setuju (5)	tidak ada, Sistem informasi gym sudah bagus
19/06/2026 22:20:56	bulan gloria	< 20 Tahun	Perempuan	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Sangat Setuju (5)	· Sistem Notifikasi Otomatis · Keamanan Data Pengguna · Laporan dan Analisis Data
21/06/2026 21:46:58	Phiraa	20–30 Tahun	Perempuan	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Setuju (4)	-
23/06/2026 13:57:46	Donine Dremansyah	> 40 Tahun	Laki - Laki	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	
23/06/2026 13:58:07	Bayu Nurmansyah	> 40 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Setuju (4)	Setuju (4)	Sangat Setuju (5)	Usahakan bisa mendukung pembayaran online
23/06/2026 13:58:37	Chandra Haditama	> 40 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Sistem informasi Gym berbasis *Website* sudah bag
23/06/2026 13:59:52	Syunasia	> 40 Tahun	Laki - Laki	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Tetap tingkatkan pelayanan by syatem

23/06/2026 16:22:47	Reza Sulaeman	20–30 Tahun	Laki - Laki	Sangat Setuju (5)	Sangat Setuju (5)	Sangat Setuju (5)	Setuju (4)	Sangat Setuju (5)	Sangat Setuju (5)	Sistem yang lebih mudah di gunakan oleh uaernya

UNIVERSITAS ATMA JAYA YOGYAKARTA